Robotic Manipulation for Granular Materials

Kegen Yu

B.Sc. Changchun (China)

Graduate Diploma Shandong (China)

October 1998

A thesis submitted for the degree of Master of Engineering of The Australian National University

Department of Engineering
Faculty of Engineering and Information Technology
The Australian National University

Declaration

This thesis contains no material which has been previously accepted for the award of any other degree or diploma at any university, institute or college, and contains no material previously published or written by another person, except where due reference is made.

Canberra, October 1998.

Kegen Yu
Department of Engineering
Faculty of Engineering and Information Technology
The Australian National University
Canberra ACT 0200, AUSTRALIA.

Conference Paper:

[C1] Kegen Yu and Jon Kieffer, "Robotic force/velocity control for following unknown contours of granular materials," submitted to the 1999 IEEE International Conference on Robotics and Automation(Detroit, Michigan, USA, May 1999).

[C2] Kegen Yu and Jon Kieffer, "Robotic force control for following unknown contours of granular materials," *Proceedings of the IASTED International Conference on Control and Applications*, Honolulu, Hawaii, USA, August 12-14, 1998.

[C3] Jon Kieffer, Kegen Yu, Junmei Cao and K. Maw, "Toward blind grouping: an approach to manipulation in unknown sensor-hostile environments," *Proceedings of the*

International Conference on Field and Service Robotics, pp.449-455, Canberra, December, 1997.

Abstract

In this thesis, we deal with three main problems: proposing and verifying a model characterising force interactions between the spherical tool and the pile of granular materials; developing a control algorithm to realize tool movement over the contour of the pile of gravel; and proposing an indirect adaptive control to tackle the uncertainties in the model parameters.

By analysing the experimental data obtained by letting the tool push the pile with shallow depth, we propose an interaction model structure. This approximate model structure is quite simple since it has only two parameters. By use of least-square identification method, we get the estimate of these two parameters. This model can be used to estimate the contact depth and the surface normal online. The experimental results demonstrate that the estimated depth and surface normal are reliable.

Then we develop a contour following algorithm which uses force feedback to adjust the desired normal force. Both force measurements and position information are utilised to estimate the surface normal and tangential online. A velocity-based position control system is used to move the tool along different planar contour shapes when the model parameters are known. Many simulations and experiments are done with various controller parameters and model parameters. The simulation and experimental results using SCARA robot illustrate the effectiveness of the proposed control algorithm.

Finally, considering the uncertainties in the model parameters, we introduce an indirect adaptation mechanism into the control algorithm. The model parameters are estimated explicitly on line. Simulation and experimental results demonstrate that the proposed adaptive method is feasible.

Acknowledgements

I would like to thank my supervisor, Jon Kieffer, for all of his efforts, encouragement, support and his appointing me as his teaching assistant over the last year. I would also like to thank Matt James who appointed me as his teaching assistant. And I would like to thank all of the people in the Department of Engineering who helped me and made it possible for me to complete my thesis in one year. Finally, I would like to thank my family so much who have given me in terms of education, encouragement, moral and financial support.

Contents

List of Figures

Notation

d	depth variable
D	depth constant
$\mathbf{e_f}$, $\mathbf{e_g}$, $\mathbf{k}$	unit vectors
$\mathbf{f}$	interaction force
f_n	normal force
f_N	normal force constant
f_t, f_τ	tangential force
f_x, f_y	Cartesian force components
k	normal stiffness
k_f, k_p	gain constant
$\mathbf{n}$	normal vector
$\mathbf{p}$	position vector
V_c,V	tangential velocity constants
v_f	velocity along $\mathbf{f}$
v_g	velocity normal to $\mathbf{f}$
v_x, v_y	Cartesian velocities
α	included angle between $\mathbf{f}$ and τ
Δt	time increment
Δx	x direction position increment
Δy	y direction position increment
λ	forgetting factor
β, γ, δ, ξ, ζ	angle variables
μ	tangential kinematic coefficient of friction
τ, $\mathbf{t}$	tangential vectors

θ_1, θ_2	joint angles
ϑ	included angle between $\mathbf{f}$ and $\mathbf{n}$

Abbreviations

CCW	counter clockwise
CFA	contour following algorithm
CFF	constant forgetting factor
DAC	direct adaptive control
DOF	degree of freedom
IAC	indirect adaptive control
RLS	recursive least squares
SFA	surface following algorithm
TVFF	time-varying forgetting factor

Chapter 1

Introduction

1.1 Background

Autonomous cleaning in Australian heavy industry, particularly in the steel-making industry, is important. One example is that at present, BHP Steel spends about 8 million dollars per annum cleaning spilled materials in the materials area of the Port Kembla Steel works alone. The cleaning work is concentrated on the removal of the granular material from under and around plant equipment such as conveyor belts. It is generally done by hand, using shovels and scrapers, exposing workers to heavy manual labour in dusty and unhealthy conditions. So developing automated cleaning systems can potentially not only decrease costs for heavy industry in Australia and over the world, but also free the workers from hostile and unhealthy environments. This research is motivated by the development of such autonomous cleaning systems.

The environment is hostile to many sensing facilities due to atmospheric dust, variable lighting, acoustic and electromagnetic noises. We believe that force sensing will play an important role because it has the advantages of being well-developed, reliable, economical, rugged and insensitive to noise above.

To remove the granular material by use of force sensing, three main problems must be coped with: perceiving the granular material, i.e., distinguishing it from other objects, such as a pillar, building walls and equipment shells; understanding the interaction properties between the tool and granular materials; and moving the robot tool on the surface while performing the cleaning tasks. A vacuum cleaner will be attached to this spherical tool. The tool moves(slides) on the surface of a pile of granular materi-

als and the vacuum cleaner cleans the materials simultaneously. The first problem is beyond this research. Our work concentrates on the last two problems.

1.2 Organisation of this Thesis

This thesis is organised as follows. In chapter 2 a force interaction model is proposed to describe the interaction between the robot tool and the gravel. Also model verification is given in this chapter. In chapter 3 a contour following algorithm is derived and a velocity-based position control system is utilised to realize the movement of robot tool over different planar contour shapes. Simulation and experimental results are also presented to demonstrate the effectiveness of the proposed control algorithm. In chapter 4 an adaptation mechanism is proposed to be introduced into the control algorithm in chapter 3. This is an indirect adaptive control (IAC) which explicitly identifies the model parameters online. Simulation and experimental results are presented to show the performance of the IAC. Each of the three chapters includes a literature review. A particularly comprehensive literature review is included in chapter 4. After tuning the velocity controller, we present some experimental results related to the direct velocity control approach in section 5. Finally conclusions are drawn in chapter 6.

Chapter 2

A Model for Interaction with Granular Material

2.1 Introduction

We are interested in manipulating a robot tool to follow the contours of an unknown pile of granular materials. The purpose is to clean granular materials effectively when an autonomous robotic cleaning system performs a cleaning task in heavy industry. To investigate this problem we have chosen to research the force-motion interaction between a spherical tool and a pile of granular materials(for our case, it is gravel). Granular materials are ubiquitous in our daily life. A pile of gravel is only one example. They consist of conglomeration of visible particles of different dimensions. However, since each configuration of granular materials has its own characteristics, the reproducibility of granular behaviour is poor [20]. The problem how to model granular materials frustrates many physicists and engineers [23]. We are not concerned with the problem of how to purely model the granular material. What we try to do is to obtain some properties of a given pile of gravel when a robot end-effector interacts with it. We will approximate a model to characterise the interaction so that the contour following control can be implemented effectively. Section 2 gives a concise description of the experimental system. An empirical model structure is proposed in section 3. Section 4 gives parameter estimation results. Model verification is presented in section 5. Finally, conclusions are drawn.

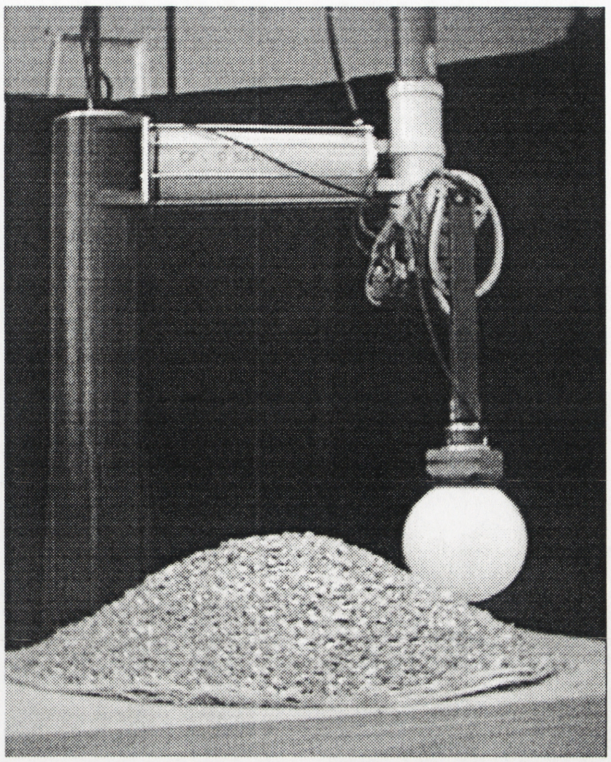


Figure 2-1: *Experimental system setup*

2.2 Experimental System

As shown in Figure 2-1, the experimental system comprises an industrial CF-310 SCARA robot, equipped with a 200 mm diameter spherical end-effector. We choose this special tool because we expect that its size and geometry will simplify the problem. The robot is a typical industrial robot with four degrees of freedom(DOF). We only use two joints (links), base and elbow, for our experiment. Both base and elbow are equipped with an encoder and a tachometre. The robot is driven by a controller(a position controller or a velocity controller) running on VME hardware under the VxWorks operating system.

For force sensing, a JR3 six-axis force torque sensor is used to obtain force data. The sensor is mounted between the tool and the wrist. This sensor measures x, y and z forces and moments (in force sensor frame), although we only use x and y force components. The frequency of sampling both force and position is 300 Hz.

The experimental object is a pile of gravel in a variety of shapes. The main components of the gravel range in size from 0.5 to 1.5 cm.

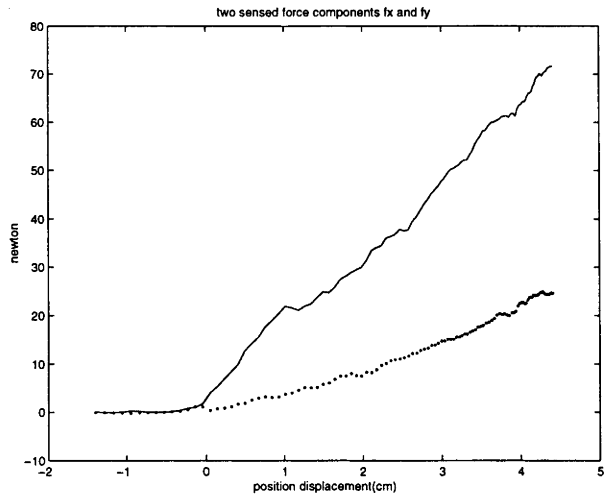


Figure 2-2: Two sensed force components f_x(solid) and f_y(dotted). Positive horizontal coordinates correspond to contact depth.

2.3 Model Structure from Qualitative Experiments

As we mentioned previously, because of the rich dynamics of granular materials, it is almost impossible to precisely model all kinds of granular materials, or even a particular pile. Our objective is to propose a feasible approximate model. A simple model can simplify controller design and a complex model will make the controller difficult to implement. So it is preferable to get a suitable simple model to describe the interactions between granular materials and the robot tool. We expect to use a low-order model. For example, if a first-order model is used, then stiffness and damping properties of the gravel should be determined. With this in mind we conducted some initial experiments to identify a simple but appropriate model.

First we moved the tool to repeatedly push the same area of the pile with different velocities under velocity control. The tool can be considered to be in rectilinear motion when it pushes the pile surface horizontally. The purpose is to determine the damping property of the gravel. By analysing the experimental results, we find out that the contribution of the velocity of the tool to the interaction forces is very small. So the damping properties can be ignored. In addition, regardless of the moving direction, the sensed forces of x and y components are approximately linear to the contact depth as shown in Fig. 2-2. Notice that we only use data from the beginning contact point to a shallow depth. When the tool pushes the pile near the surface normal, the sensed normal force is also approximately linear to the depth. The sensed tangential force

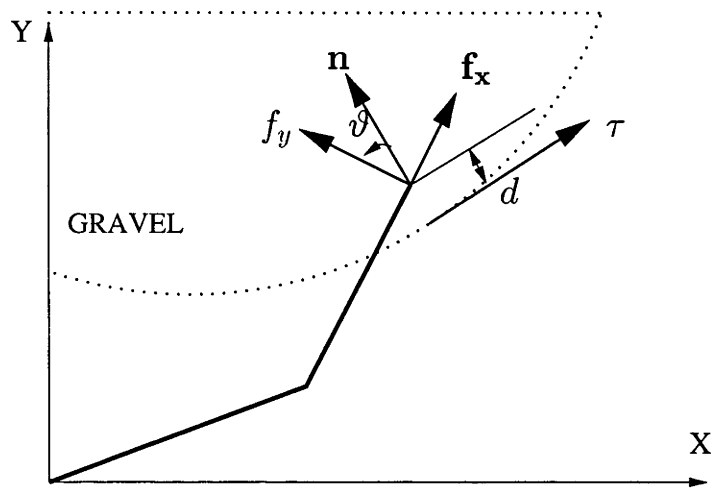


Figure 2-3: Sensed force($\mathbf{f_x}, \mathbf{f_y}$), normal and tangential forces($\mathbf{f_t}, \mathbf{f_n}$), reference coordinates(X, Y).

is quite small. This implies that when robot tool pushes a pile of gravel, the normal direction interaction could be considered as a spring with a constant stiffness.

The tangential force is treated as arising from Couloumb friction which is the product of normal force and friction factor when a mass moves on a plane. We also moved the tool along a sloping plane of the gravel with about the same depth. The sensed tangential force was found to be relatively constant for a constant depth. Of course, if the depth is very small, the sensed forces will be quite noisy. For simplicity, the tangential force is taken to be proportional to position displacement in the normal direction. So an empirical model structure is proposed as follows

$$\mathbf{f_n} = k\,d \tag{2.1}$$

$$\mathbf{f_t} = \mu\,d\,sign(\dot{\mathbf{t}}) \tag{2.2}$$

where $\mathbf{f_t}$ and $\mathbf{f_n}$ are the tangential and normal forces which the tool applies, $\mathbf{f_n}$ is normal force , d is the contact depth, k is stiffness constant, μ is coefficient of kinematic friction and $\dot{\mathbf{t}}$ is the velocity in the tangential direction(τ). Notice that the force sensor frame is along the elbow as shown in Figure 2-3.

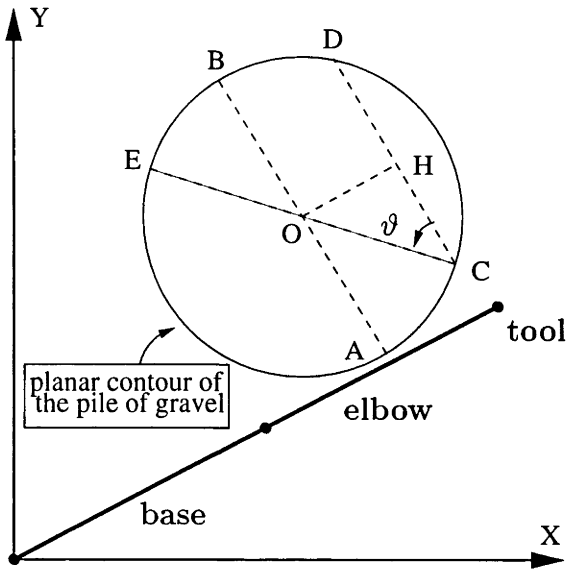


Figure 2-4: Determination of ϑ

2.4 Model Parameters Identification

Experiments were then conducted to identify values for the two constants k and μ using least squares techniques.

In order to make the two constants μ and k represent the general properties of the pile of gravel, we let the tool push the pile from different directions with different values of ϑ. Seven groups of data are obtained. Each data set was visually inspected to identify the instant of contact and truncated to isolate the data associated with shallow-depth interactions. The actual depth d was estimated based on the displacement from the time of first contact and knowledge of the actual value of ϑ. To make the experimental results as reliable as possible, the shape of the pile of gravel was made to be almost a cone, especially with a precise circular base. The two joints(base and elbow) are always aligned such that the movement of the tool can be consider to be linear when the distance is only about a few centimetres as shown in figure 2-4. For shallow-depth contact, we can assume that the tool will move along line CD after it touches point C because that is the direction of the tool velocity before contact. So ϑ can be approximately determined by the following equation.

$$\vartheta = arc\,sin\,(OH/OC) \tag{2.3}$$

Note that the lengths of line OC(radius) and OH can be approximately got in advance.

The least squares method was then applied to estimate values using the following formulation.

From Figure 2-3 and applying force equilibrium, we have

$$\begin{cases} f_t = f_x \cos\vartheta - f_y \sin\vartheta \\ f_n = f_x \sin\vartheta + f_y \cos\vartheta \end{cases} \tag{2.4}$$

We also can use f_t and f_n to express f_x and f_y

$$\begin{cases} f_x = f_t \cos\vartheta + f_n \sin\vartheta \\ f_y = -f_t \sin\vartheta + f_n \cos\vartheta \end{cases} \tag{2.5}$$

Using equation(2.1) and (2.2), equation (2.5) can be written in matrix notation as

$$\mathbf{f_{xy}} = \varphi^T \theta \tag{2.6}$$

where

$$\mathbf{f_{xy}} = (f_x \quad f_y)^T \tag{2.7}$$

$$\theta = (\mu \quad k)^T \tag{2.8}$$

$$\varphi^T = \begin{pmatrix} d\cos\vartheta & d\sin\vartheta \\ -d\sin\vartheta & d\cos\vartheta \end{pmatrix} \tag{2.9}$$

and d is the contact depth in the normal direction.

For a given ϑ, if we get m sampling data, then equation (2.6) can be extended as

$$\mathbf{F_{xy}} = \phi^T \theta \tag{2.10}$$

where

$$\mathbf{F_{xy}} = (\mathbf{f_{xy}}^T(1) \quad \mathbf{f_{xy}}^T(2) \quad \cdots \quad \mathbf{f_{xy}}^T(m))^T \tag{2.11}$$

$$\phi = (\varphi^T(1) \quad \varphi^T(2) \quad \cdots \quad \varphi^T(m))^T \tag{2.12}$$

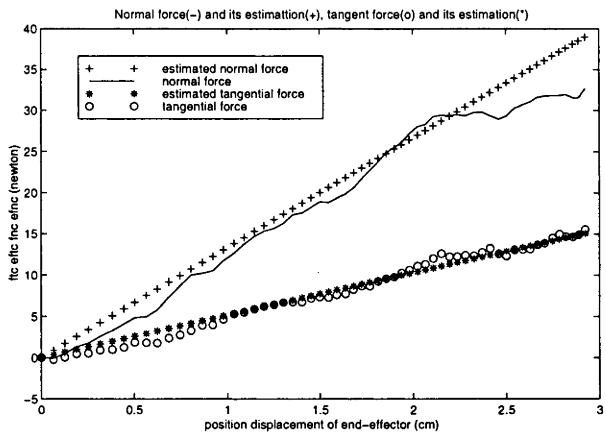


Figure 2-5: Comparison between (f_t, f_n) and $(\hat{f}_t, \hat{f}_n)$

To get a general model, we use n different ϑ values for our experiment and we have n equations in the form of equation(2.10). Grouping the n equations yields

$$\mathbf{Y_{xy}} = \Phi^T \theta \tag{2.13}$$

where

$$\mathbf{Y_{xy}} = (\mathbf{F_{xy}}^T(1) \quad \mathbf{F_{xy}}^T(2) \quad \cdots \quad \mathbf{F_{xy}}^T(n))^T \tag{2.14}$$

$$\Phi = (\phi^T(1) \quad \phi^T(2) \quad \cdots \quad \phi^T(n))^T \tag{2.15}$$

Applying least square estimate algorithm produces

$$\hat{\theta} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{Y_{xy}} \tag{2.16}$$

where $\hat{\theta}$ is the estimate of θ. Using the obtained data to solve the above equations, we obtain the estimates of μ and k

$$\hat{k} = 16.25 \ N/cm$$

$$\hat{\mu} = 5.24 \ N/cm$$

Notice that k is much lower than the tool stiffness(estimated as about 100 N/cm). This means that k is the property of the gravel pile, instead of the robot. Fig. 2-5 shows one example of comparisons between actual and estimated normal forces, and between actual and estimated tangential forces by use of $\hat{k}$ and $\hat{\mu}$ for one experiment when ϑ is about -19 degrees.

2.5 Model Verification

After the model parameter estimates are known, we would like to use the model to estimate the depth d and the angle ϑ. Using equations (2.1) and (2.2), equation (2.4) can be rewritten as

$$\begin{bmatrix} cos\,\vartheta \\ sin\,\vartheta \end{bmatrix} = \frac{d}{f_x^2 + f_y^2} \begin{bmatrix} f_x & f_y \\ -f_y & f_x \end{bmatrix} \begin{bmatrix} \mu\, sign(\dot{\mathbf{t}}) \\ k \end{bmatrix} \tag{2.17}$$

Applying the trig identity $sin^2(\vartheta) + cos^2(\vartheta) = 1$ produces

$$d = \sqrt{\frac{f_x^2 + f_y^2}{\mu^2 + k^2}} \tag{2.18}$$

If parameters k and μ are known, then equations (2.17) and (2.18) can be used to estimate d and ϑ on line. Note that equation (2.17) becomes indeterminate when the sensed forces are too small. But this case can be eliminated by a threshold criteria. Also notice that the solution of the equation (2.17) is dependent on sign($\dot{\mathbf{t}}$) which has two possible values. This ambiguity can be solved by assuming that sign($\dot{\mathbf{t}}$) is either 1 or -1, then checking the tangential velocity calculated by the resulting ϑ. If it is consistent with the assumption, then the assumption is correct. Otherwise, the alternative of sign($\dot{\mathbf{t}}$) should be used.

By choosing different ϑ values which are determined by the methods in the previous section as shown in figure 2-4, we got seven groups of data. Figures 2-6–2-7 show the best and the worst of the results. Obviously, the results illustrate that the estimated depth and the estimated angle $\hat{\vartheta}$(which tells us the surface normal) are quite near the true values. This means that the model is appropriate and acceptable.

2.6 Conclusion

In this chapter we propose an approximate model to describe the interactions between the robot tool and the gravel. This model structure is quite simple with only two parameters. By use of the least squares method, we get the two model parameter estimates. The verification results demonstrate that this model can be feasibly used to determine the depth and the local surface normal and tangential of the gravel pile.

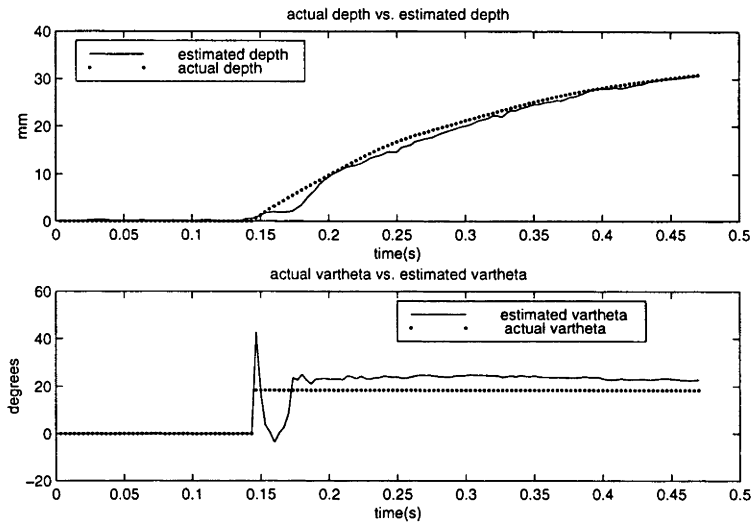


Figure 2-6: Best estimation results

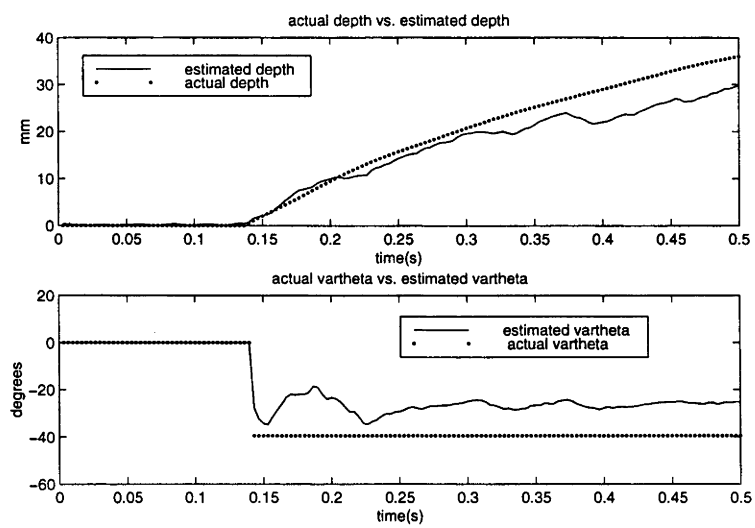


Figure 2-7: Worst estimation results

Chapter 3

Contour Following Control

3.1 Introduction

In this chapter, we focus on realizing the robot tool movement over the surface of the pile of gravel. This is novel research work, since to the author"s knowledge no similar work has been carried out previously. It is also important. When the robot system whose tool is equipped with a vacuum cleaner performs a cleaning task, the tool should move on the surface of the pile of granular materials rather than go inside the pile. Based on the proposed contour following control algorithm, we have done a lot of simulations and experiments. The results demonstrate that the robot tool can move over the different planar contours of the pile of gravel satisfactorily.

In this chapter, we first give a review on robotic surface following control. Then we present a simple contour following algorithm. Some simulation results are then given. Finally experimental results are presented to show the effectiveness of the contour following algorithm(CFA).

3.2 Literature Review

There are many applications which require the robot manipulator to interact with the environment, especially to follow the shape of the environment during performing its task. These applications include surface polishing, grinding, scraping, machining, sheep shearing, and so on. Accordingly, there are some surface following algorithms(SFA) in the literature. Based on the different environmental characteristics, two main SFA

strategies exist. One is for dealing with rigid surfaces and the other is to tackle the soft surfaces. Here soft means that the environment is deformable.

In some previous works on robot force/position control, the surface model of the object was given or obtained by various methods to allow estimation in advance[16][17]. Recognising that it is very time consuming in computing to get a global model, R.E. Goddard *et al*[14] proposed a control method for robot rolling and sliding over an unknown object. This method makes use of velocity control rather than position control. Using velocity control for robot probing only needs local information instead of a global model of the object. In addition, holonomic and non-holonomic constraints were considered for motion planning of robot fingers. But only simulations were done to show the effectiveness of the control algorithm.

In order to move a robot end-effector over the surface of an object, some information about the surface locally or globally must be obtained. Some researchers have made efforts to estimate the constraint surface for force control. Merlet[27] proposed a method which used force measurements to determine the force normal. Blauer and Belanger[3] proposed using an extended Kalman filter to estimate some unknown parameters related to the constraint surface. Kazanzides[22] made use of interaction force and end-effector velocity to determine the constraint surface normal and tangential directions. In the method proposed by T. Yoshikawa and A. Sudou[54], both force and position measurements were utilized to estimate the constraint surface locally on-line. This method dealt with the three-dimensional space directly. In addition, a strategy was taken to compensate the frictional force. Finally, experimental results using a SCARA robot were presented to show the good combination between the online estimation algorithm and the dynamic hybrid control.

If the force controller is equipped with a low-friction roller, then the measured force at the surface is mostly the normal force. Also the normal force and the tangential force have a simple relation. D. Bossert *et al*[5] presented a SFA which made use of this relation between normal and tangential forces. B. Yong[53] proposed a method which used the preview control for contour-following robot force control. The preview control uses the information about both the present contour shape and the future one. A Kalman filter algorithm is utilised to reconstruct the system state and to filter the noise from the force measurements. The author also presented experimental results by

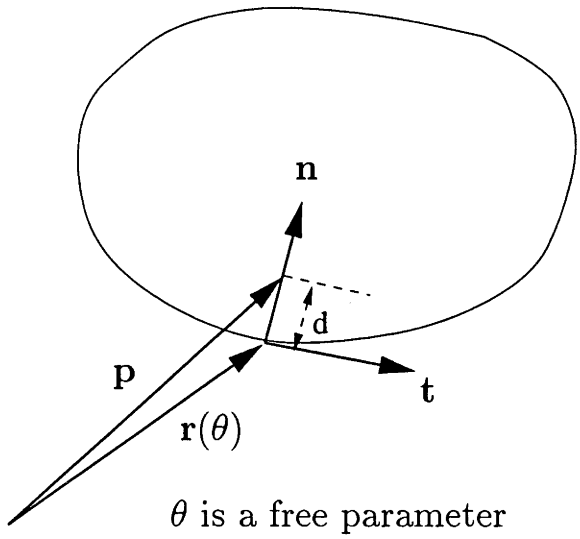


Figure 3-1: Object contour

using the PUMA industrial robot to verify the preview control. Some other surface following algorithms were proposed in[31][32][33].

However, all of the above mentioned references are based on the fact that the object surface is rigid. How to move the robot end-effector over the surface of a pile of granular materials(or gravel for our situation) is a new topic. Different from the rigid surfaces, the surface of a pile of granular materials has its own unique characteristics. One obvious aspect is that the surface is rough. another property is that when the end-effector pushes the pile, the local shape will be deformed permanently. So when the robot end-effector moves on the surface of a pile of granular materials, some materials will be pushed aside and the surface shape will be changed.

3.3 Contour Following Algorithm

In this section, we will derive a surface following control algorithm. This algorithm makes use of control velocity to adjust the moving trajectory.

Figure 3-1 shows the planar contour of a pile of granular materials. Let the parametric curve $\mathbf{r}(\theta)$ represent the boundary of the pile of gravel. Assume $|\dot{\mathbf{r}}(\theta)| \neq 0$. Taking $\tau(\theta)$ and $\mathbf{n}(\theta)$ as the unit tangential and unit normal for $\mathbf{r}(\theta)$ respectively, we have

$$\tau(\theta) = \frac{\dot{\mathbf{r}}(\theta)}{|\dot{\mathbf{r}}(\theta)|} \tag{3.1}$$

$$\mathbf{n}(\theta) = \mathbf{k} \times \tau(\theta) \tag{3.2}$$

where $\mathbf{k}$ is the unit vector that points out of the paper.
Using the model deduced in the previous chapter, the interaction force exerted on the environment by the tool can be rewritten as

$$\mathbf{f}(\theta, d) = \begin{cases} k\, d\, \mathbf{n}(\theta) + \mu\, d\, \mathbf{t}(\theta) & d \geq 0 \\ 0 & elsewhere \end{cases} \tag{3.3}$$

where d, k and μ have the same definitions as before with d, the actual depth in normal direction; k, normal direction stiffness coefficient; μ, tangential direction force coefficient.

The control objective is to move the end-effector over the planar contour of gravel with the desired normal force and tangential velocity. This aim can be expressed as

$$\begin{cases} \mathbf{f} \cdot \mathbf{n}(\theta) &= f_N\, \mathbf{n}(\theta) \\ |\dot{\mathbf{p}}| &= V_c \end{cases} \tag{3.4}$$

where f_N is the desired normal force, $\mathbf{p}$ is the position vector of the robot tool, and V_c is the desired velocity.

Naturally, we can change the tool velocity in the surface normal to adjust the normal force. So we use the following control law.

$$\dot{\mathbf{p}} = V_c\, \mathbf{t} + k_f\, (f_N - f_n)\, \mathbf{n} \tag{3.5}$$

where k_f is a gain constant, f_n is the measured normal force.

Equation (3.5) can also be written as

$$\dot{\mathbf{p}} = V_c\, \mathbf{t} + k_p\, (D - d)\, \mathbf{n} \tag{3.6}$$

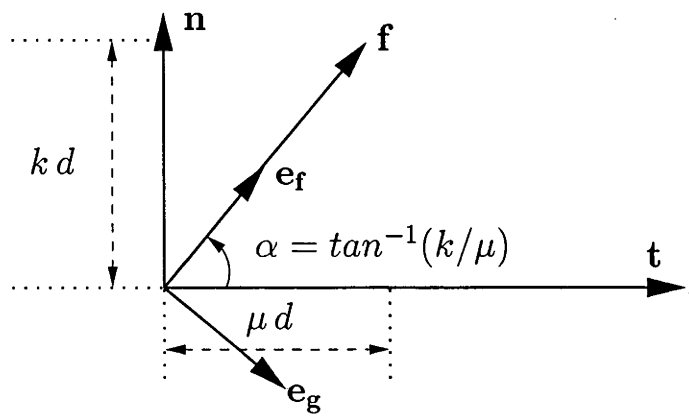


Figure 3-2: Determination of the control velocities

where k_p is a gain constant which is equal to $k_f\,k$, D is depth constant which equals f_N/k.

Let $\mathbf{e}$ be force tracking error, we have

$$\mathbf{e} = f_n\,\mathbf{n} - f_N\,\mathbf{n} \tag{3.7}$$

The above control law satisfies

$$(\dot{\mathbf{e}} + k_f\,k\,\mathbf{e}) \cdot \mathbf{n} = 0 \tag{3.8}$$

Proof:

$$\begin{aligned}
(\dot{\mathbf{e}} + k_f\,k\,\mathbf{e}) \cdot \mathbf{n} &= \dot{f_n}\mathbf{n} + k\,k_f(f_n - f_N) \\
&= k\dot{d}\,\mathbf{n} + k\,k_f(f_n - f_N) \\
&= k(\dot{\mathbf{p}} - \dot{\mathbf{r}}(\theta))\mathbf{n} + k\,k_f(f_n - f_N) \\
&= k\,\dot{\mathbf{p}}\,\mathbf{n} + k\,k_f(f_n - f_N) \\
&= k\,V_c\,\mathbf{t}\,\mathbf{n} + k\,k_f(f_N - f_n) + k\,k_f(f_n - f_N) \\
&= 0
\end{aligned}$$

The control law(3.5) is a little awkward to implement because the computation related to the coordinate ($\mathbf{t}$, $\mathbf{n}$) will be relatively complex. Now we give a more convenient approach to compute the velocity $\dot{\mathbf{p}}$. Figure 3-2 shows the relationship among the

sensed force, the normal and the tangential. $\mathbf{e}_f$ and $\mathbf{e}_g$ are unit vectors. $\mathbf{e}_f$ is along the force direction and $\mathbf{e}_g$ is normal to $\mathbf{e}_f$. Our purpose is to factorise the velocity $\dot{\mathbf{p}}$ into two velocity components, i.e.v_f and v_g. They are in the same directions as unit vectors $\mathbf{e}_f$ and $\mathbf{e}_g$ respectively. The velocity components (v_f, v_g) are referenced to coordinate system ($\mathbf{e}_f$, $\mathbf{e}_g$) which is easy to determine by:

$$\begin{cases} \mathbf{e}_f &= \mathbf{f}/|\mathbf{f}| \\ \mathbf{e}_g &= -\mathbf{k} \times \mathbf{e}_f \end{cases}$$

where $\mathbf{k}$ has the same definition as in equation(3.2).

From figure 3-2, we get

$$\begin{aligned} \dot{\mathbf{p}} &= (\dot{\mathbf{p}} \cdot \mathbf{e}_f)\,\mathbf{e}_f + (\dot{\mathbf{p}} \cdot \mathbf{e}_g)\,\mathbf{e}_g \\ &= (V_c\,\mathbf{t} + k_f\,(f_N - f_n)\,\mathbf{n}) \cdot \mathbf{e}_f\,\mathbf{e}_f + (V_c\,\mathbf{t} + k_f\,(f_N - f_n)\,\mathbf{n}) \cdot \mathbf{e}_g\,\mathbf{e}_g \\ &= (V_c\,\mathbf{t} \cdot \mathbf{e}_f + k_f\,(f_N - f_n)\,\mathbf{n} \cdot \mathbf{e}_f)\,\mathbf{e}_f + (V_c\,\mathbf{t} \cdot \mathbf{e}_g + k_f\,(f_N - f_n)\,\mathbf{n} \cdot \mathbf{e}_g)\,\mathbf{e}_g \\ &= (V_c\,cos\,\alpha + k_f\,(f_N - f_n)\,sin\,\alpha)\,\mathbf{e}_f + (V_c\,sin\,\alpha - k_f\,(f_N - f_n)\,cos\,\alpha)\,\mathbf{e}_g \end{aligned}$$

Assume that v_f and v_g are two velocity components along $\mathbf{e}_f$ and $\mathbf{e}_g$, respectively. Then we have

$$\begin{pmatrix} v_f \\ v_g \end{pmatrix} = \begin{pmatrix} cos\,\alpha & sin\,\alpha \\ sin\,\alpha & -cos\,\alpha \end{pmatrix} \begin{pmatrix} V_c \\ k_f\,(f_N - f_n) \end{pmatrix}.$$

Substituting $f_n = |f|\,sin\,\alpha$ into the above equation yields

$$\begin{pmatrix} v_f \\ v_g \end{pmatrix} = \begin{pmatrix} cos\,\alpha & sin\,\alpha \\ sin\,\alpha & -cos\,\alpha \end{pmatrix} \begin{pmatrix} V_c \\ k_f\,(f_N - |f|\,sin\,\alpha) \end{pmatrix}. \tag{3.9}$$

Obviously, v_f and v_g are functions of $|\mathbf{f}|$ only which can be measured online. So this control law is much easier to implement than control lay(3.5). Note that only α which equals $tan^{-1}(k/\mu)$, not k or μ, actually appears in the above control law.

Now let us determine how the error in $\hat{\alpha}$ as shown in figure 3-3 affects the system performance such as the errors in the normal force and tangential velocity. For simplicity, we assume that the circumstance of the planar contour is infinite and the robot is an ideal robot which can carry out the velocity and position commands perfectly.

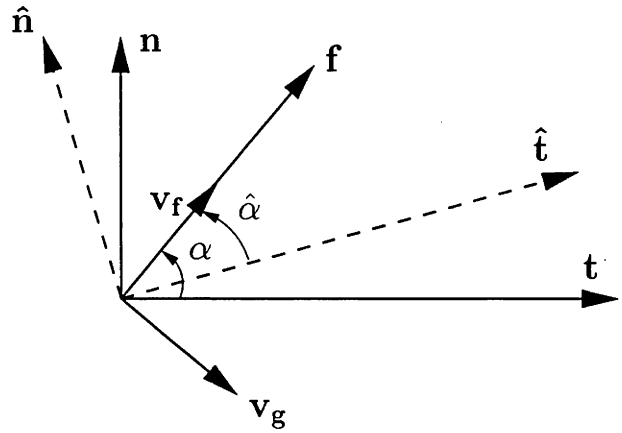


Figure 3-3: Determination of the force and velocity tracking errors

Under these conditions, this robot system will quickly reach the steady state after the CFA is executed. When the system is in the steady state, the tangential velocity is constant and the normal velocity of the tool is zero. So we have

$$\begin{pmatrix} v_f \\ v_g \end{pmatrix} = \begin{pmatrix} cos\,\alpha & sin\,\alpha \\ sin\,\alpha & -cos\,\alpha \end{pmatrix} \begin{pmatrix} V_t \\ 0 \end{pmatrix} = \begin{pmatrix} cos\,\alpha \\ sin\,\alpha \end{pmatrix} V_t \tag{3.10}$$

where V_t is the actual tangential velocity.

In addition, as the estimated α is used, the control law becomes

$$\begin{pmatrix} v_f \\ v_g \end{pmatrix} = \begin{pmatrix} cos\,\hat{\alpha} & sin\,\hat{\alpha} \\ sin\,\hat{\alpha} & -cos\,\hat{\alpha} \end{pmatrix} \begin{pmatrix} V_c \\ k_f\,(f_N - \hat{f}_n) \end{pmatrix}. \tag{3.11}$$

where $\hat{f}_n$ is the estimated normal force.

Combining equation(3.10) and (3.11) yields

$$\begin{cases} cos\,\hat{\alpha}\,V_t + k_f\,sin\,\hat{\alpha}\,\hat{f}_n = cos\,\hat{\alpha}\,V_c + k_f\,f_N\,sin\,\hat{\alpha} \\ sin\,\hat{\alpha}\,V_t - k_f\,cos\,\hat{\alpha}\,\hat{f}_n = sin\,\hat{\alpha}\,V_c - k_f\,f_N\,cos\,\hat{\alpha} \end{cases} \tag{3.12}$$

Solving the above equation produces

$$V_t = \frac{V_c}{cos(\hat{\alpha}-\alpha)}$$
$$\hat{f}_n = f_N - \frac{V_c}{k_f} tan(\hat{\alpha} - \alpha)$$

Using $f = \hat{f}_n / sin(\hat{\alpha})$, where f is the resultant force, we finally have

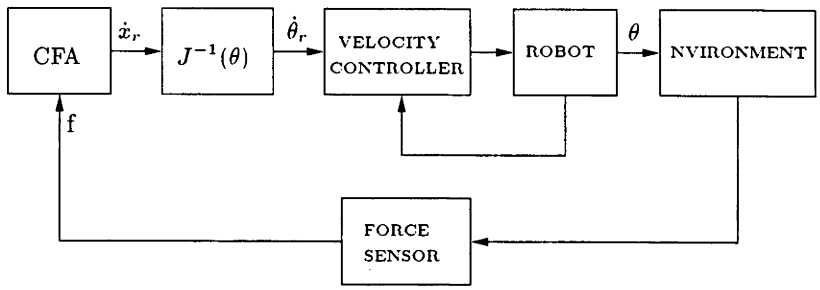


Figure 3-4: Block scheme of velocity control

$$\begin{cases} V_t = \frac{V_c}{cos(\hat{\alpha}-\alpha)} \\ f_n = \{f_N - \frac{V_c}{k_f} tan(\hat{\alpha}-\alpha)\} sin\,\alpha / sin\,\hat{\alpha} \end{cases} \tag{3.13}$$

where f_n is the actual normal force magnitude.

So we can approximately predict the force and velocity tracking errors provided that we know the error of the estimated α.

3.4 Control System Structure

In the previous section, we derived a contour following algorithm using force feedback for velocity compensation. In this section we explain how this algorithm can be applied to a real robot.

The most natural way is to use the algorithm to drive a robot under close-loop velocity control as shown in figure 3-4. Another way is to use it to drive a robot under velocity-based position control as shown in figure 3-5 and figure 3-6. In addition, we can imagine more sophisticated approaches to control. For example, the manipulator dynamic model can be used for the control objective, but we choose not to consider them unless necessary.

Comparatively the system shown in figure 3-4 is simpler. The close-loop velocity control can be implemented directly. But the velocity controller of our robotic system has not been satisfactorily implemented.

As for the system shown in figure 3-5, it is worth noticing that the actual position of the robot end-effector at each sampling point is not considered. The determination of the next position completely depends on the integration of the two reference velocities. When the difference between the actual position and the desired one increases gradually,

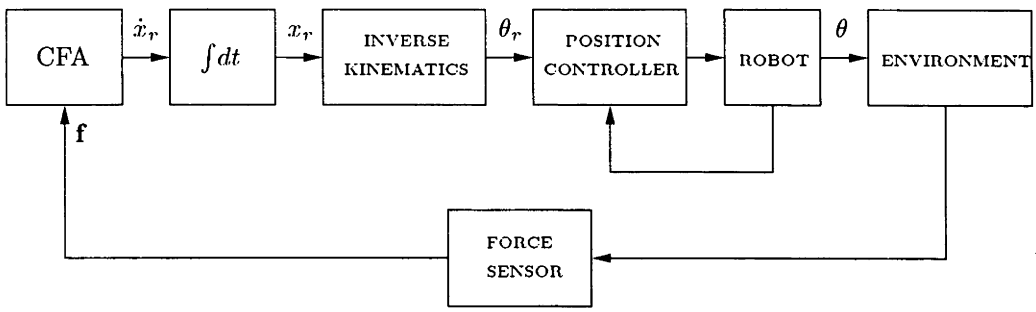


Figure 3-5: Block scheme one of velocity-based position control

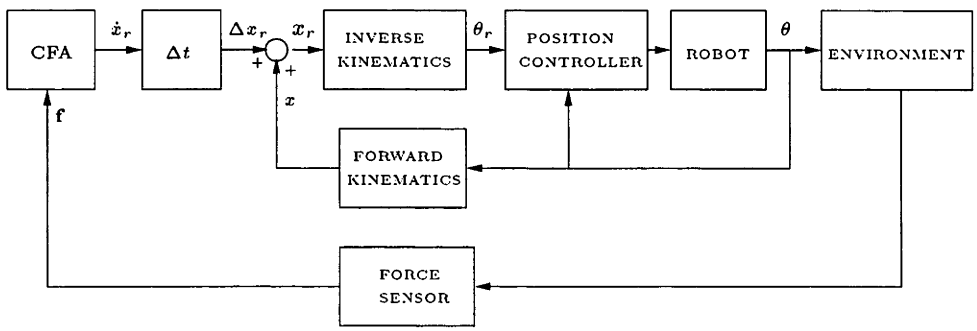


Figure 3-6: Block scheme two of velocity-based position control

the tracking will fail due to lose of contact, or too great a depth. Our approach to solve this problem is to use the present position information as well as the control velocities for the computation of the next position of the end-effector. The idea is illustrated in figure 3-6.

Notice that the proposed controllers are driven solely by force measurements. The surface normal and tangential are computed by force measurements as well as position information. Also notice that the controllers do not include a dynamic model of the manipulator. We have chosen to implement the contour following control with a simple velocity-based position control. It is expected that the performance would be improved with a model-based robot motion controller(e.g. computed-torque). But we have found that it is not necessary to use a sophisticated controller to get good results.

Now let us use figure 3-7 to determine the next position the end-effector should move to from the present point.

From figure3-7, we have

$$\delta = \beta + \theta_1 + \theta_2 - pi/2 \tag{3.14}$$

where $\beta = atan2(f_y, f_x)$, f_y and f_x are two measured force components.

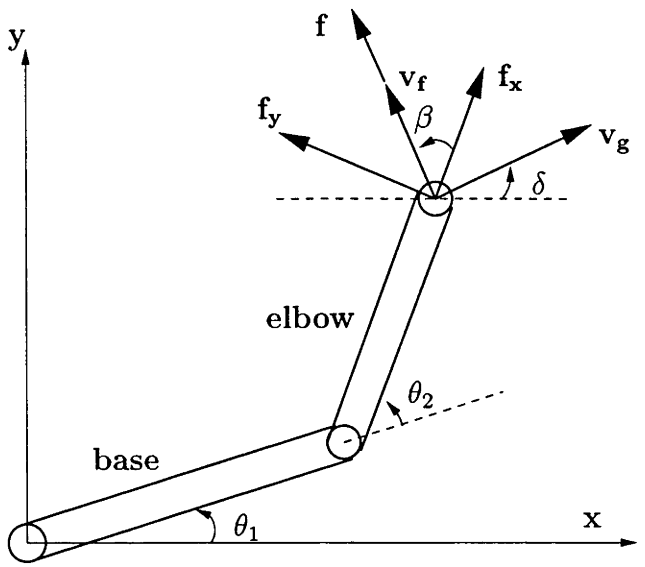


Figure 3-7: Computation of new position using control velocities

Then we have the relationship between the velocities(v_f, v_g) and (v_x, v_y).

$$\begin{cases} v_x = cos\,\delta\, v_g - sin\,\delta\, v_f \\ v_y = sin\,\delta\, v_g + cos\,\delta\, v_f \end{cases} \tag{3.15}$$

where v_x and v_y are two next desired Cartesian velocities.

Furthermore, the Cartesian position increments are

$$\begin{cases} \Delta x = v_x \Delta t \\ \Delta y = v_y \Delta t \end{cases} \tag{3.16}$$

where Δt is the time increment.

Finally, we obtain the next position

$$\begin{cases} x_{new} = x_{old} + \Delta x \\ y_{new} = y_{old} + \Delta y \end{cases} \tag{3.17}$$

Applying the inverse kinematics, we can transform the Cartesian position into joint posit-on to carry out the joint space position control.

3.5 Digital Filtering

When the robot tool moves on the surface of the pile of gravel, the interaction force is very noisy. Fortunately, the JR3 force sensor has six built-in cascade filters with

highest and lowest cutoff frequencies of $500Hz$ and $0.4883Hz$ respectively. We choose the built-in filter with cutoff frequency of $31.2Hz$. This selection is based on the fact that the actual sampling frequency of both position and force in our experiments is about 10 to 20 Hz. Also it is desirable to attenuate the noises as much as possible. However, we find that there still exist rich noises in the filtered force measurements with any built-in filter. So it is desirable to design a digital filter to smooth the trajectory of the end-effector and furthermore to suppress some high frequency noises on-line.

Filtering is used to extract signals from noise. If the signal and noise spectra are not overlapping, it is possible to design a filter to pass the desired signals and to remove the unwanted noises. If the signal and noise fields are partly overlapping, it is expedient to suppress the noises without attenuating the signals largely.

There are three basic classes of digital filters. They are fast transform filters, nonrecursive filters and recursive filter. Fast transform filters implement the generalised frequency response of the digital filter using fast transform algorithm. The FFT filters are the best known of such filters. Recursive and nonrecursive filters realize the frequency response $H(z)$ in hardware using delays, scalers and summers, or in microprocessor hardware. These filters can also be implemented in software by use of sets of recursion equations. Recursive filters are often referred to as infinite impulse response(IIR) filters. IIR filters normally require less hardware and can carry out filtering tasks more quickly than finite impulse response(FIR) filters often referred to as nonrecursive filters. However, nonrecursive filters are inherently stable. They can also be designed to have a linear phase property in the passband.

For simplicity, we design a FIR filter using windowing and truncation technique [4]. This filter is used to remove some high frequency noises from the force measurements. It can also smooth the trajectory of the end-effector. Now let us consider a filter whose input-output behaviour is described with the following difference equation.

$$y(k) = a\,x(k) + b\,x(k-1) + c\,x(k-2) \tag{3.18}$$

Here designing a filter means to determine the coefficients a, b and c. Consider the frequency response function $H(e^{j\omega T})$ for an ideal lowpass filter whose cutoff frequency is ω_c as shown in Fig.7.

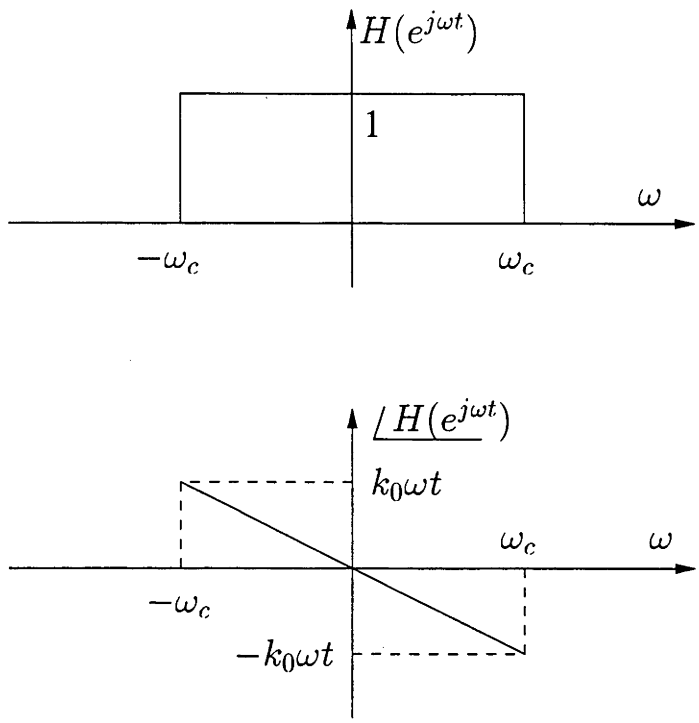


Figure 3-8: Lowpass filter characteristics

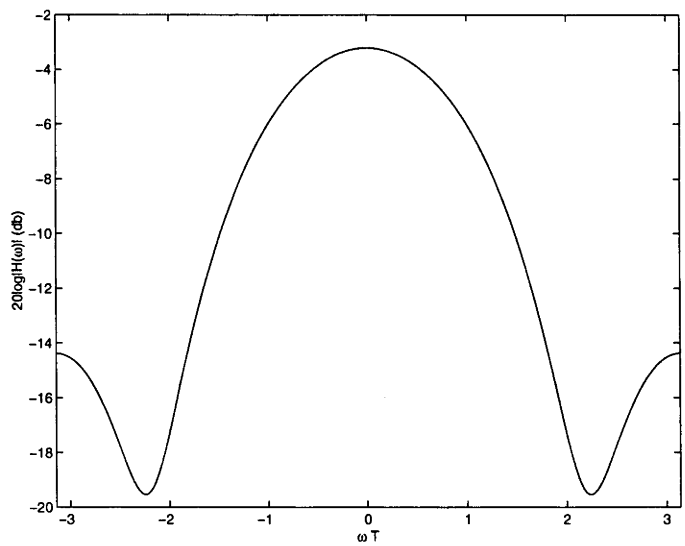


Figure 3-9: Magnitude of the frequency response of the designed filter

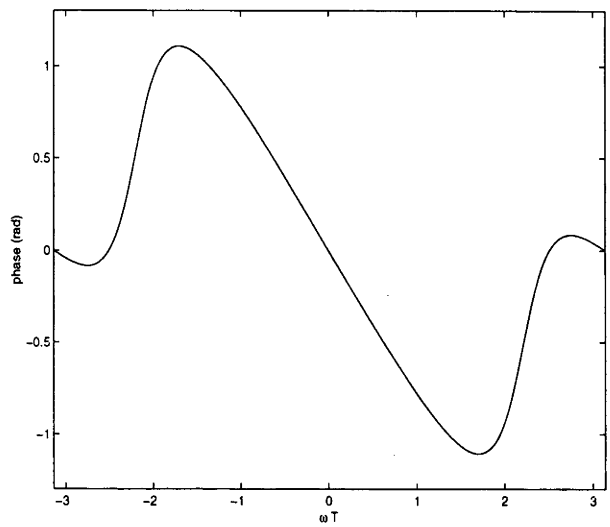


Figure 3-10: Phase of the frequency response of the designed filter

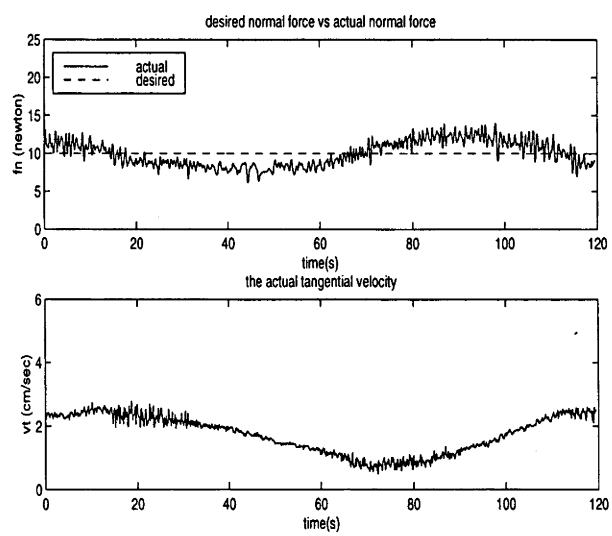


Figure 3-11: Experimental results(circular contour)using the designed filter

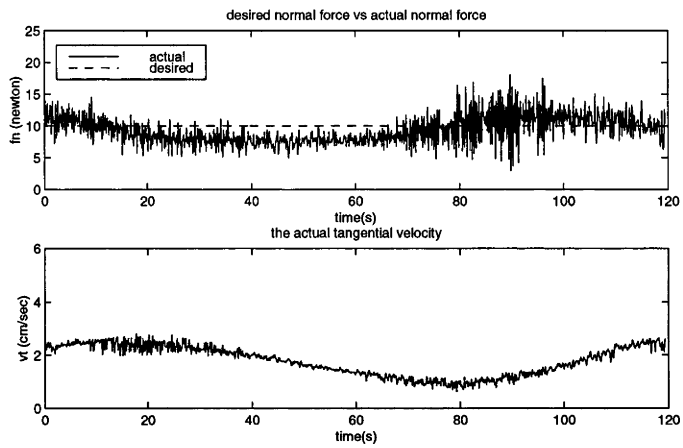


Figure 3-12: Experimental results(circular contour)without using the designed filter

Using Fourier series expansion for $H(e^{j\omega T})$ yields

$$H(e^{j\omega T}) = \sum_{k=-\infty}^{\infty} h(k)\, e^{-jk\omega T}$$

Then the impulse response function can be calculated by

$$h(k) = \begin{cases} \frac{sin[(k-k_0)\,\omega_c T]}{(k-k_0)\,\pi} & k \neq k_0 \\ \\ \frac{\omega_c T}{\pi} & k = k_0 \end{cases} \tag{3.19}$$

where k_0 is a constant. For brevity, we take k_0=0. If $\omega_c = 9Hz$ and $T = 0.1\,s$, then $a = h(0) = 0.286$, $b = h(1) = 0.25$ and $c = h(2) = 0.155$. The actual magnitude and phase of the frequency response of the designed filter are shown in figure 3-9, 3-10. The frequency roll-of of this filter is about 108 db/decade.

Figure 3-11 shows the results from one experiment using the designed digital filter. The controller parameters are $k_f = 0.4\,cm/s/N$, $\Delta t = 0.5\,s$ and $T = 1/7.5\,s$. The $\hat{\alpha}$ is taken as 0.85 rad. For comparison, the results without using the designed filter are also given in Figure 3-12. It is obvious that the designed filter does have its merit. Since the filter gain is less than 1, it also attenuates the useful signals. So it is necessary to amplify the filtered signals or to increase the reference value by, say, ten percent. Note that even though the force is decreased and the force phase is also changed, the force is much more smooth. In addition, we found that it is much easier for the tracking to be fail without a filter. So it is preferable to use the filter.

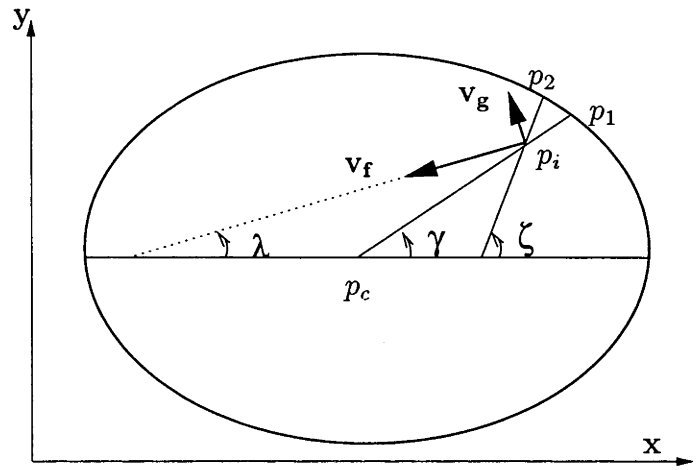


Figure 3-13: Simulation using elliptical contour

3.6 Simulation Results

In this section, we first give some simulation results by use of an ideal virtual robot. Then the simulation results using a real robot are presented. There exists a difference between these two simulations. The ideal virtual robot can perform a positioning task without any error. But the actual robot normally cannot reach the desired position exactly, particularly when the time interval is very small. So it is necessary to present both simulation results. The second case is actually a hybrid combination of simulation and experiment. The robot is real and motion data is experimental.

3.6.1 Simulation Using Virtual Ideal Robot

In this subsection, we assume that the robot can perform tracking task without any positioning error. In order to deal with the general situation, we use an ellipse as the planar contour of a pile of gravel as shown in figure 3-13.

Assume that $p_c(x_c, y_c)$ is the ellipse canter, point p_i is the present position of the tool, and p_1 and p_2 are two points on the contour. In order to compute the sensed force at p_i, we first determine the equation of the normal at p_2 which goes through p_i. To simplify computation, we approximate the slope of the normal at p_2 to the slope of the normal at p_1. This should not create much error if the depth is shallow. Using the known coordinates $p_c(x_c, y_c)$ and $p_i(x_i, y_i)$ produces

$$\gamma = atan2\,(y_i - y_c, x_i - x_c) \tag{3.20}$$

Then the coordinate at p_1 can be calculated by

$$\begin{cases} x_1 = a\cos\gamma + x_c \\ y_1 = b\sin\gamma + y_c \end{cases} \tag{3.21}$$

where a and b are major and minor axes. According to the properties of ellipses, the equation of the normal at p_1 is

$$\frac{(x - x_c)\,a^2}{x_1 - x_c} - \frac{(y - y_c)\,b^2}{y_1 - y_c} = a^2 - b^2 \tag{3.22}$$

Equation(3.22) can be rewritten as

$$y = A\,x + B \tag{3.23}$$

where

$$A = a^2/b^2\,tan(\gamma)$$

$$B = y_c - (x_c + x_1(a^2/b^2 - 1))\,tan(\gamma)$$

So the equation of the normal at p_2 is

$$y = A\,x + C \tag{3.24}$$

where $C = y_i - A\,x_i$. The equation of the ellipse is

$$\frac{(x - x_c)^2}{a^2} + \frac{(y - y_c)^2}{b^2} = 1 \tag{3.25}$$

Substituting equation (3.24) into equation (3.25) yields

$$M\,x^2 + N\,x + L = 0 \tag{3.26}$$

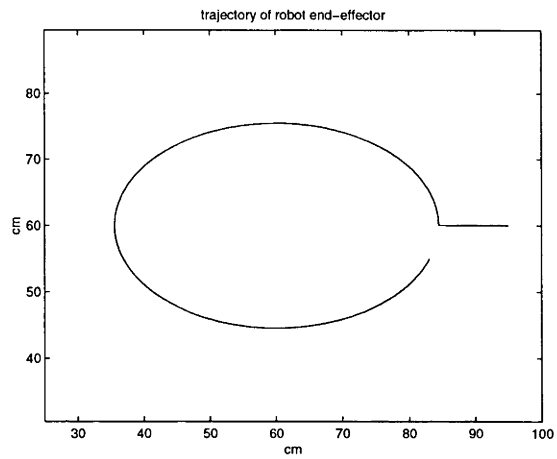


Figure 3-14: Simulation using exact model parameters

where

$$M = b^2 + (a\,A)^2$$

$$N = -2(b^2\,x_c + a^2\,A y_c - C))$$

$$L = (b\,x_c)^2 + a^2(y_c - C)^2 - (a\,b)^2$$

Solving equation (3.26) for x_2 and then solving equation (3.24), we obtain the coordinate $p_2(x_2, y_2)$. Now the depth can be calculated with

$$d = \sqrt{(x_2 - x_i)^2 + (y_2 - y_i)^2)}$$

Notice that there are two solutions for x_2 and y_2. Accordingly, d has two values. The larger value should be discarded. Using $f_n = k\,d$ to compute f_n and then using equation (3.9) to calculate v_f and v_g, finally, we can determine the new position by

$$\begin{cases} x_{new} = x_{old} - v_f\,cos\,\lambda\,\Delta t - v_g\,sin\,\lambda\,\Delta t \\ y_{new} = y_{old} - v_f\,sin\,\lambda\,\Delta t + v_g\,cos\,\lambda\,\Delta t \end{cases} \tag{3.27}$$

where $\lambda = atan(A) + \alpha - \pi/2$, α is a known constant which is the angle between the contact force and the surface tangential.

Choosing some control parameters such as k_p and Δt is important. In the following subsection, there will be a detailed discussion about how to choose these parameters. Here we choose k_f=0.5 cm/s/N and Δt=0.2 s. Also considering the actual situation,

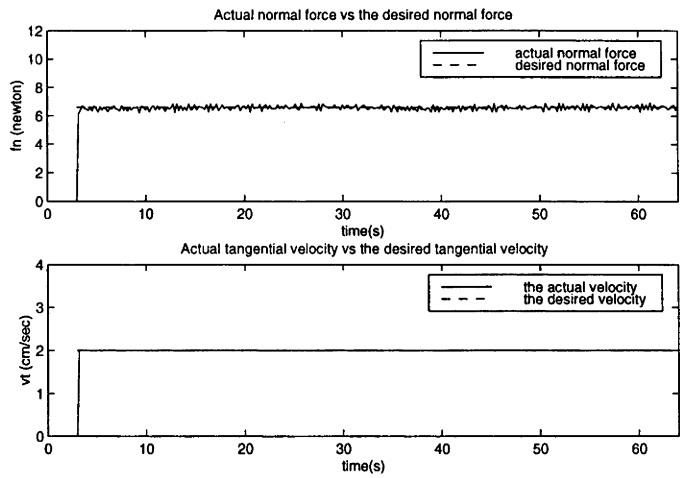


Figure 3-15: Simulation using exact model parameters

we add random noise to the force signals. This noise is random numbers which are produced by the standard C *rand* function. The magnitude of these random numbers is between 0.2 Newton and -0.2 Newton. The random noise is also used in the following subsection.

When the actual and estimated parameters are assumed to be the same with k(normal stiffness coefficient)=12.0 N/cm and μ(tangential force coefficient)=10.55 N/cm, the simulation results are shown in Figures 3-14–3-15. The corresponding true α is 48.68 degree. Figures 3-16–3-17 show the simulation results when the estimated model parameters are different from the real ones with K_e=15.2 N/cm and μ_e=8.2 N/cm such that the estimated $\hat{\alpha}$ is 61.65 degree. When this estimate of alpha is used, the computed normal force and tangential velocity tracking errors by equation(3.13) are 0.052 cm/sec and 1.67 Newton, respectively. The simulation results accord with the computed results. So when the exact model parameters are used, the actual normal force and the tangential velocity will be almost the same as what are desired. On the other hand, if the estimated parameters are not accurate. the actual normal force and the tangential velocity will be different from the desired ones. However, if the estimated model parameters are not too much different from the true values, the actual normal force and tangential velocity are still near the desired values.

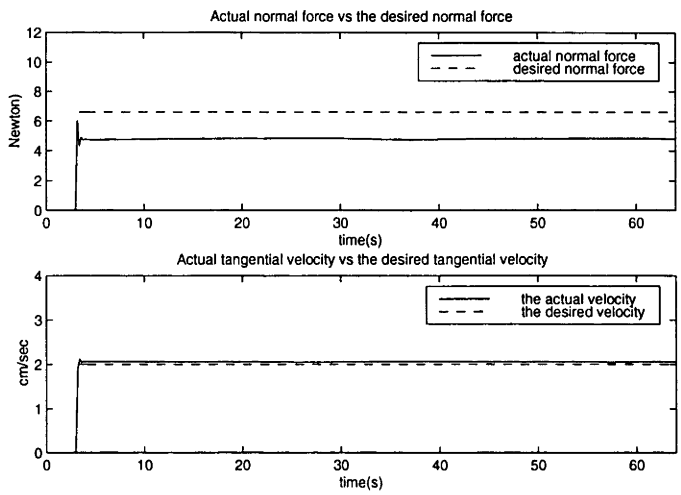


Figure 3-16: Simulation using estimated model parameters(without noises)

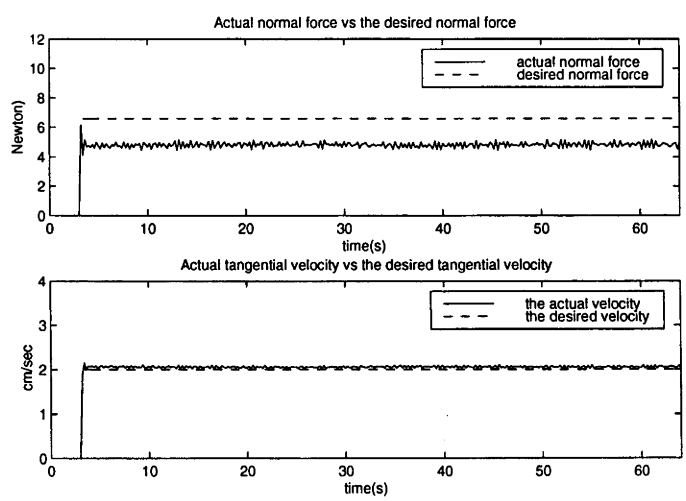


Figure 3-17: Simulation using estimated model parameters(noises added)

3.6.2 Hybrid Simulation Using Real Robot

The simulation in the above subsection can be treated as performed by use of an ideal robot to execute point-to-point movement without any error. Now we use the SCARA robot to realize the desired point-to-point movement. Here the real robot is interacting with a simulation of the pile of gravel. All of the model parameters are the same as in the above subsection with $k = 12.0\,N/cm$, $\mu = 10.55\,N/cm$ and $\alpha = 0.85\,rad$.

Let the sampling period be T. The sampling frequency can be set up by frequency splitting. For example, the sampling frequency 20Hz can be produced by dividing the servo rate(300 Hz) by 15. Then the actual Cartesian velocities are

$$\begin{cases} v_x(i) = \Delta x(i)/T \\ v_y(i) = \Delta y(i)/T \end{cases} \tag{3.28}$$

where $\Delta x(i)$ and $\Delta y(i)$ are actual Cartesian position displacements at the ith sampling point. Notice that the time increment Δt and T are treated the same in the previous section when the ideal virtual robot is used. But for actual situation, we can treat them separately to get better control results. Here Δt is used to compute position increment by use of the known control velocity components v_f and v_g. If we use T to calculate the Cartesian position increment with $\Delta x = v_x T$ and $\Delta y = v_y T$, the increment will be very small. This will result in poor tracking performance, or even tracking failure with the real robot. We found that the tool velocity will decrease quickly until near zero and there will be a large error in force tracking too. Normally, Δt should be much larger than T. The following results will demonstrate that the appropriate Δt is about 0.9 s and T is about 0.1 to 0.05 s.

Using the estimated tangential direction, we can obtain the actual tangential velocities by use of the Cartesian velocities. Note that the mentioned tangential velocity here is different from the desired velocity V_c. Throughout all of the simulations and experiments in this thesis, V_c is equal to 2cm/sec. V_c is the desired tangential velocity when the velocity closed-loop system is used. The velocity feedback closed-loop system tries to adjust the output velocity to the given value. Since the position control is used, the velocity will be dependent on the sampling period T and the time increment Δt as well as gain constant k_f. Large time increments will bring about large velocity. The later simulation and experimental results will demonstrate that even though the

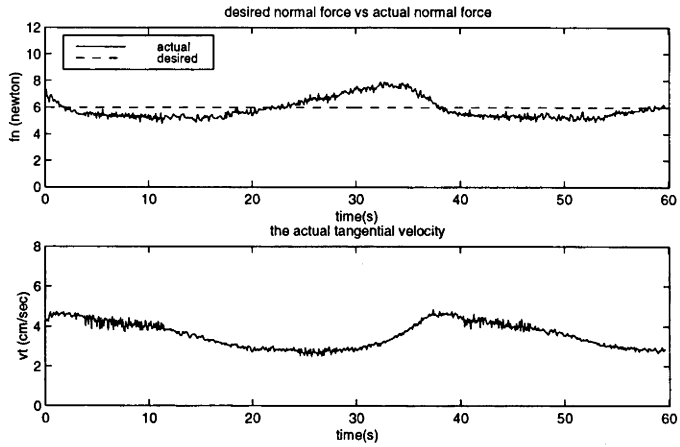


Figure 3-18: Simulation results with $k_f = 0.4cm/s/N, \Delta t = 1.0s$ and $T = 0.1s$

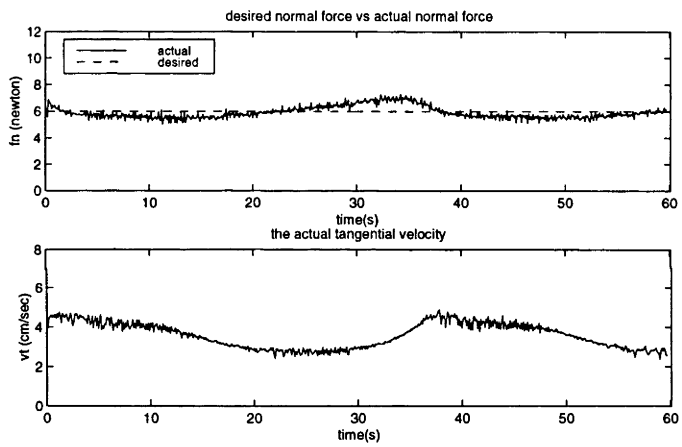


Figure 3-19: Simulation results with $k_f = 0.7cm/s/N, \Delta t = 1.0s$ and $T = 0.1s$

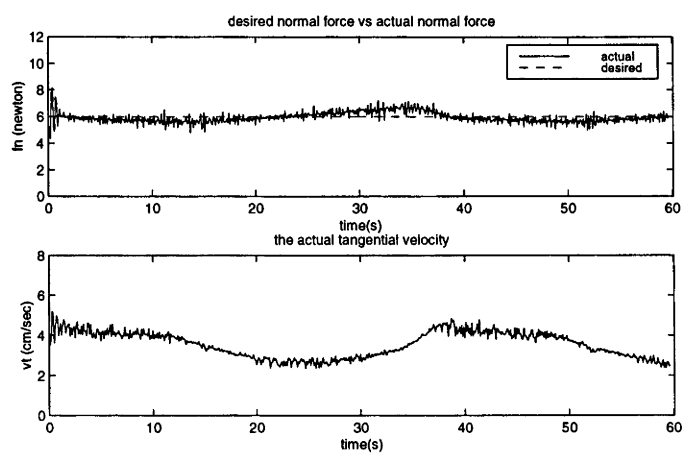


Figure 3-20: Simulation results with $k_f = 1.0cm/s/N, \Delta t = 1.0s$ and $T = 0.1s$

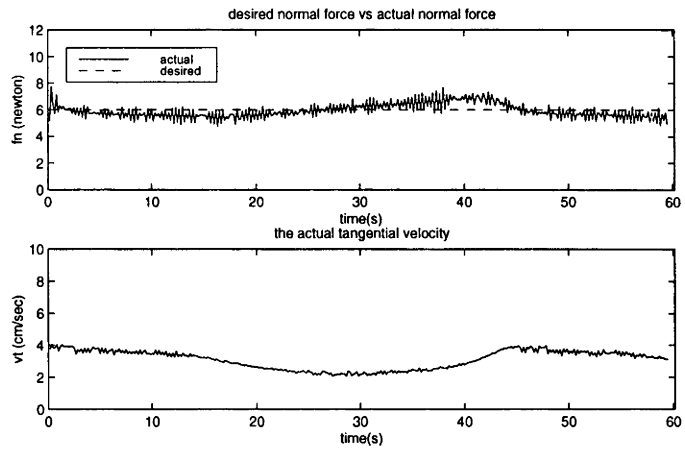


Figure 3-21: Simulation results with $k_f = 0.76cm/s/N, \Delta t = 0.9s$ and $T = 1/6s$

position control approach is used, the resultant tangential velocities will be near the desired values if Δt, T and k_f are appropriately chosen.

Choosing control system parameters is important in the system design. Control performance will be greatly affected by the controller parameters which are k_f, Δt and T for our case.

Figures 3-18–3-20 illustrate the simulation results when three different gain constants, i.e. 0.4, 0.7 and 1.0 cm/s/N are used. The sampling frequency and time increment are 10 Hz and 1.0 s respectively. Theoretically, large gain k_f will increase control sensitivity and adjustment. But very large gain will also result in large oscillation and even unstable. On the other hand, the adjustment performance will be poor with small gain. The results shown in Figures 3-18–3-20 tally with the theory. For the given Δt and T, the suitable gain constant k_f is between 0.7 and 1.0.

Then we determine what is the suitable sampling frequency. The results with four different frequencies are shown in Figures 3-21–3-23. The gain k_f and time increment Δt are 0.76 cm/s/N and 0.9 s respectively. When k_f and Δt are the given values, the appropriate sampling frequency is between 10 and 20 Hz. When sampling frequency is over 150 Hz, the contour following control will fail. On the other hand, when the frequency is lower than 6Hz, there exist large oscillations.

Furthermore we use different time increment to understand its effect upon the system performance. The results are illustrated in Figures 3-25-3-28.

Obviously, larger time increment will bring about better tracking performance. However, too big time increment will be too sensitive to error as shown in Figure 3-28.

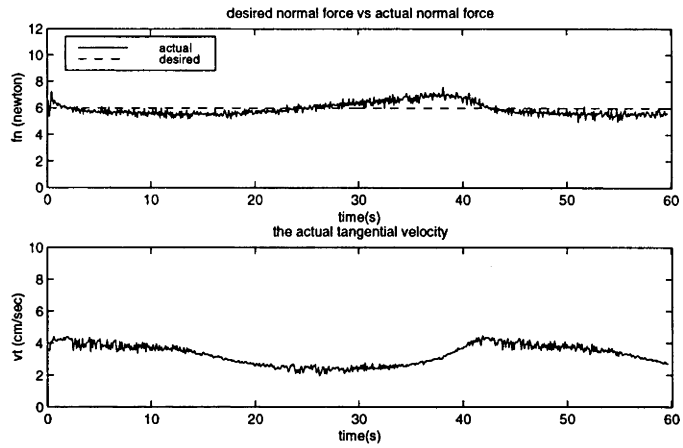


Figure 3-22: Simulation results with $k_f = 0.76cm/s/N, \Delta t = 0.9s$ and $T = 0.1s$

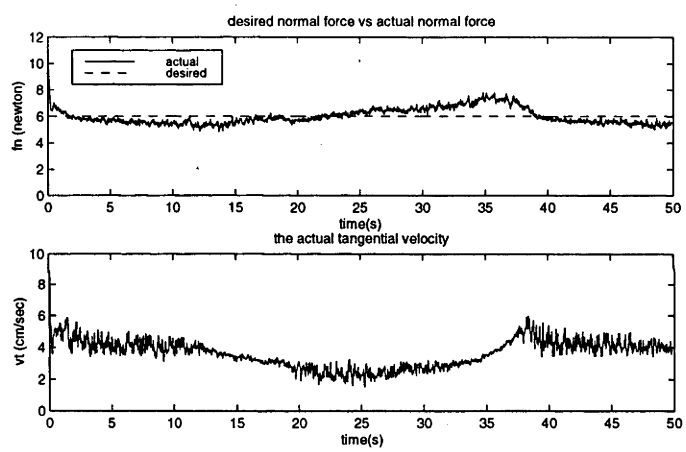


Figure 3-23: Simulation results with $k_f = 0.76cm/s/N, \Delta t = 0.9s$ and $T = 1/30s$

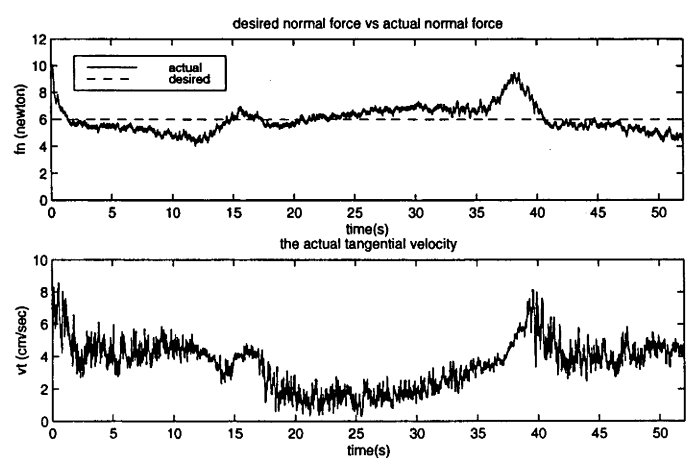


Figure 3-24: Simulation results with $k_f = 0.76cm/s/N, \Delta t = 0.9s$ and $T = 1/75s$

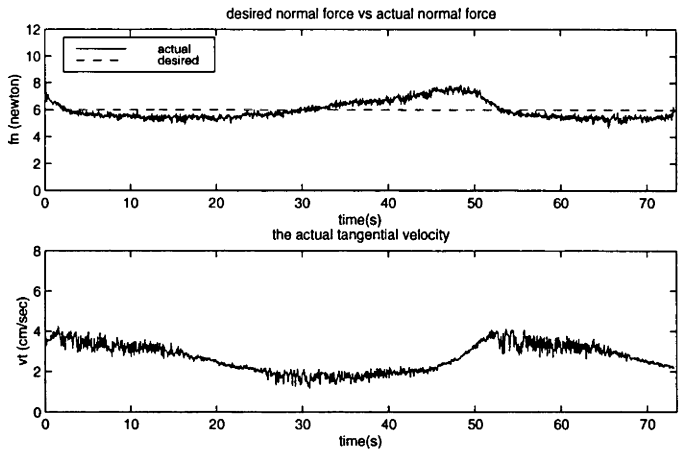


Figure 3-25: Simulation results with $k_f = 0.7cm/s/N, \Delta t = 0.7s$ and $T = 1/15s$

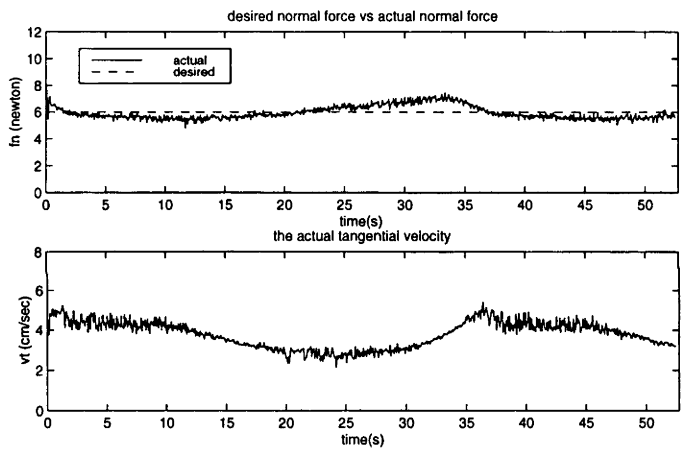


Figure 3-26: Simulation results with $k_f = 0.7cm/s/N, \Delta t = 1.0s$ and $T = 1/15s$

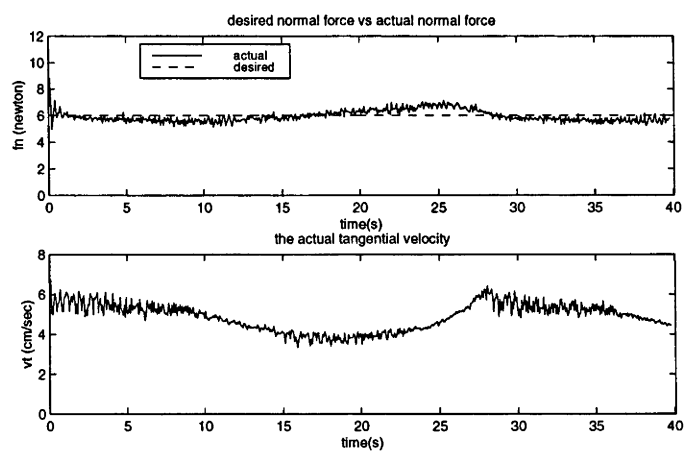


Figure 3-27: Simulation results with $k_f = 0.7cm/s/N, \Delta t = 1.3s$ and $T = 1/15s$

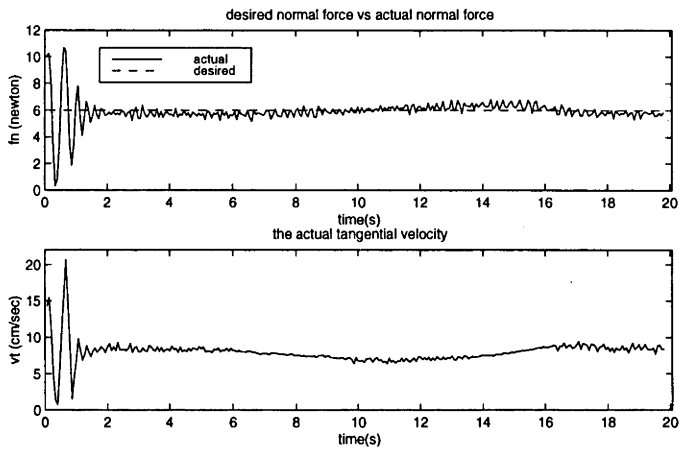


Figure 3-28: Simulation results with $k_f = 0.7cm/s/N, \Delta t = 2.2s$ and $T = 1/15s$

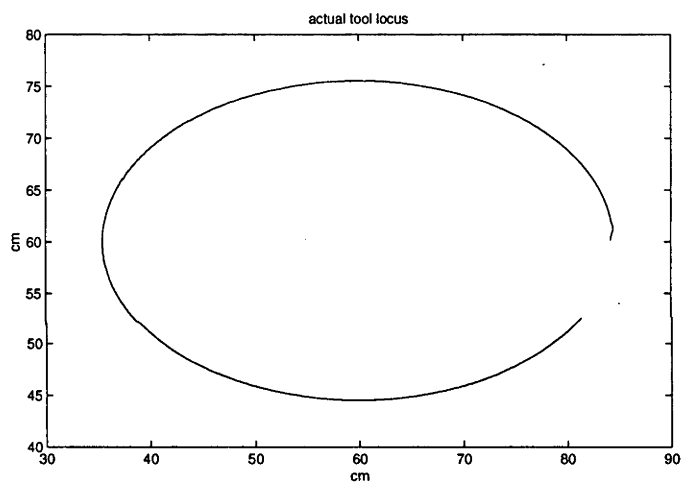


Figure 3-29: Simulation results with $k_f = 0.7cm/s/N, \Delta t = 1.0s$ and $T = 1/15s$

It will easily result in tracking failure in an actual situation. It is wise to choose time increment around 1 s.

Figure 3-29 shows the actual tool locus when $k_f = 0.7\,cm/s/N, \Delta t = 1.0\,s$ and $T = 1/15\,s$. Since the depth is much smaller compared to the elliptical major or minor axis, the tool trajectory is still smooth. Because the tool locus for other cases are almost the same as this one, it is not necessary to show them.

Notice that there exist positioning error in joint space of both base and elbow which is about between -0.005 rad and 0.005 rad. This error will bring about errors between -0.5 cm and 0.5 cm in Cartesian space. But because of the force feedback adjustment, the positioning error does not largely affect the force tracking performance.

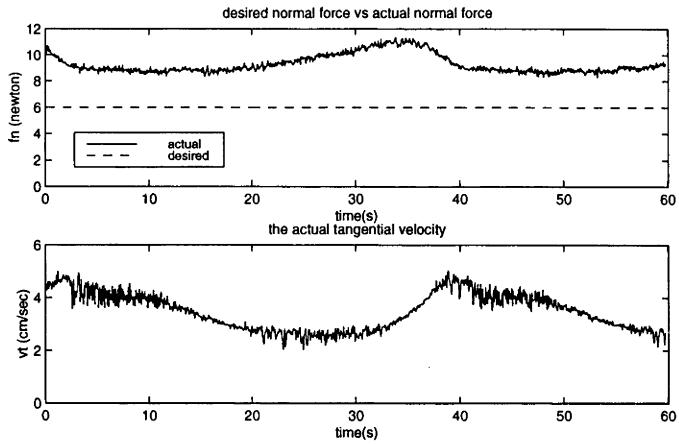


Figure 3-30: Simulation results with $\hat{\alpha} = 0.4\,rad$

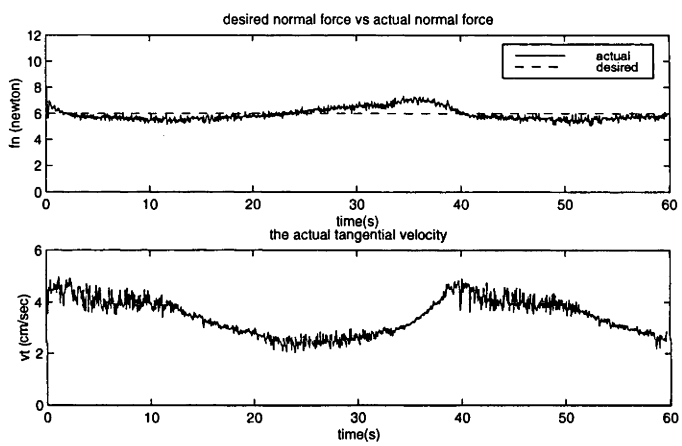


Figure 3-31: Simulation results with $\hat{\alpha} = 0.85\,rad$

All of the above simulations using the SCARA robot to interact with the simulated pile of gravel are done with the assumption that the estimated parameters in the interaction model are the same as the actual ones. Now let us have a look at what will happen when the two parameters are different.

Figures 3-30–3-32 give the simulation results with three different α values, i.e. 0.4, 0.85 and 1.26 rad. The true α is taken as 0.85 rad. Clearly, when $\hat{\alpha}$ is less than the true value, the normal force output will be larger than the desired value. For this case, the force tracking error is about 50 percent. Contrarily, when $\hat{\alpha}$ is larger than the true value, the actual normal force will be less than the desired one. The tracking error now is about 20 percent. Of course, when $\hat{\alpha}$ equals the real α, the actual normal force best matches the desired value. The results also tally with the theoretical computation.

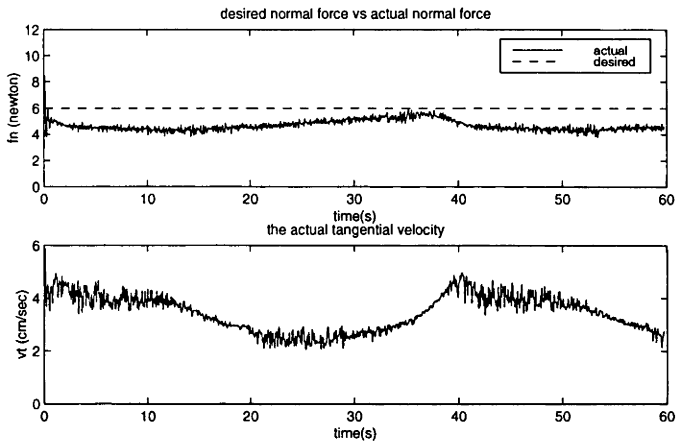


Figure 3-32: Simulation results with $\hat{\alpha} = 1.26\,rad$

3.7 Suggestion for Dealing with Two Special Cases

When the tool moves along the contour, two phenomena will probably happen due to the abrupt change of the environmental parameters and/or the pile shape. One is that the tool touches the pile with too small depth and then loses of contact with the pile surface. The other is that the depth is too big so that this may cause damage to the equipment. Normally, if the estimated model parameters are applicable and the controller parameters are appropriate, the second problem will not happen since the CFA will reduce the contact force when the force is much larger than the desired value. For safety, when the sensed force is too big, the robot will be forced to stop. When the sensed force is too small to provide reliable surface information, the tool may lose contact with the pile contour. If the tool completely loses contact with the surface, one simple method can be used to recover the tool movement over the pile surface. It is to let the tool retouch the surface again based on the previous position information. For example, a routine controls the tool to move toward the pile using a recent estimate of the normal $\mathbf{n}$ at a specified low speed. After the sensed force reaches the desired value, the CFA goes into effect again.

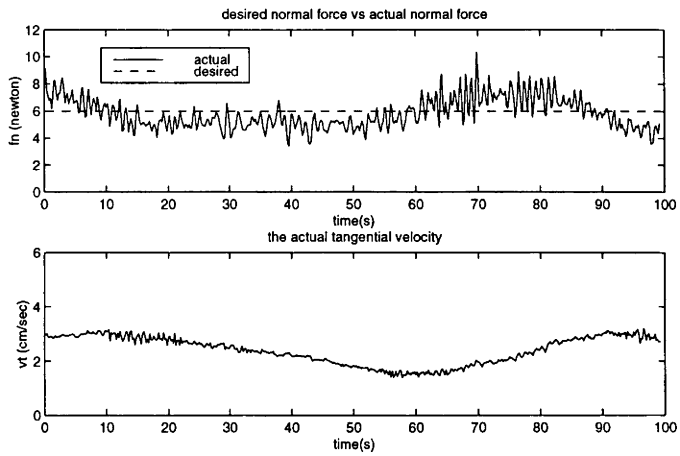


Figure 3-33: Experimental results(circular contour) with $k_f = 0.5\,cm/s/N$, $\Delta t = 0.7\,s$ and $T = 0.2\,s$

3.8 Possible Impact of the Vacuum Cleaner upon the Control Algorithm

As we mentioned previously, the aim of the final system is to clean the granular materials. A vacuum cleaner will be attached to the robot tool. As the robot tool moves on the surface of the pile of granular materials, the vacuum cleaner removes the materials. Under these practical circumstances, the contour following algorithm will be inevitably affected due to the additional force(e.g. suction) and the material removed by the robot as it moves. The impact of these factors upon the sensed and required forces may be complex. For example, it depends on the position of the attached vacuum cleaner and the whole structure of the tool. There may exist three approaches to deal with this problem. The first method is to incorporate the sucking action into the interaction model. So the model describes the total dynamic interaction process. The second approach is to add this impact into the control algorithm. For instance, some compensation can be utilized to eliminate the effect of the sucking process. A third approach is to use the contour-following algorithm without vacuuming to identify the shape of the pile. Then apply a vacuuming algorithm until the pile shape needs to be reidentified. The techniques developed in this thesis are intended to be a foundation for developing a complete system. They do not in themselves provide a complete solution.

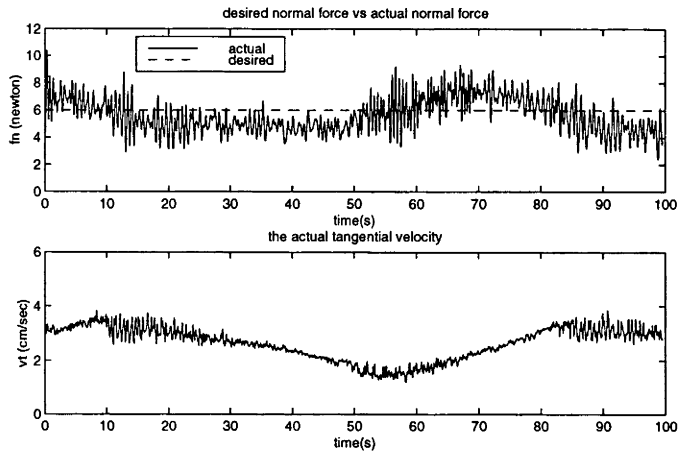


Figure 3-34: Experimental results(circular contour) with $k_f = 0.5\,cm/s/N$, $\Delta t = 0.7\,s$ and $T = 0.1\,s$

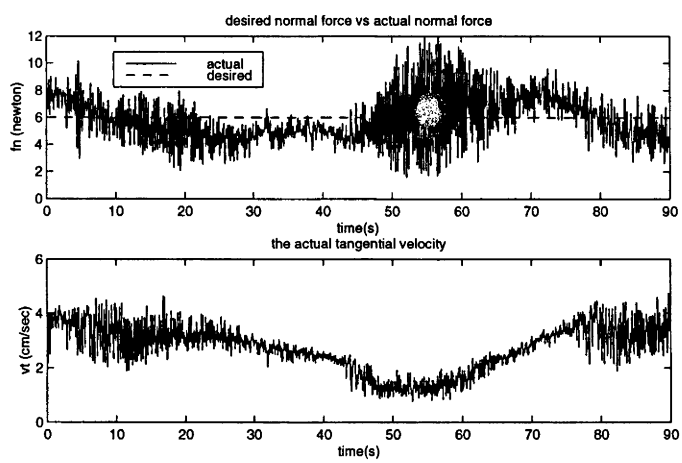


Figure 3-35: Experimental results(circular contour) with $k_f = 0.5\,cm/s/N$, $\Delta t = 0.5\,s$ and $T = 1/30\,s$

3.9 Experimental Results

After verifying the effectiveness of the control algorithm in simulations, we use the same previous pile of gravel for some real experiments. First the pile is in a conical shape which has an approximate circular planar contour. The model parameters are a little different from those used for simulations. By analysing the experimental results, we think the actual α is around 0.5 rad. Remember that alpha is the angle between the surface normal and the contact force vector. When the feedback variable is the normal force, the most important model parameter is $\hat{\alpha}$. Similarly, we use different control parameters in our experiments to compare the results to choose the most suitable control parameters.

When the time increment equals 0.7 s and the sampling frequency is 5 Hz, 10 Hz and 30 Hz respectively, the experimental results are shown in Figure 3-33–3-35. It is impressive that the system works well with the sampling frequency 5 Hz. On the other hand, when the frequency is higher than 30 Hz, the force tracking ability seems to be poor.

To have a reliable judgement of how the controller parameters affect the system performance, we apply the integral square-error(ISE) criterion. According to ISE criterion, the quality of system performance can be evaluated by the following integral:

$$\int_0^\infty e^2(t)\,dt$$

For discrete data, the above equation can be written as

$$\sum_0^\infty e^2(i)$$

By calculating the summation, the error performance indexes(450 sampling points) are 570, 765 and 1325 N^2 corresponding to frequency 5, 10 and 30 Hz, respectively. So the appropriate sampling frequency is around 10 Hz.

Next the frequency of 10 Hz is chosen and the time increment is changed from 0.5 through 0.7 to 0.9 s. The results are illustrated in Figures 3-36–3-38. It is clear that larger time increment tries to increase the force tracking performance. But it also brings about larger vibrations. Also the tangential velocity of the tool increases with the time increment. Applying the ISE criterion, the performance indexes(650 sampling points)

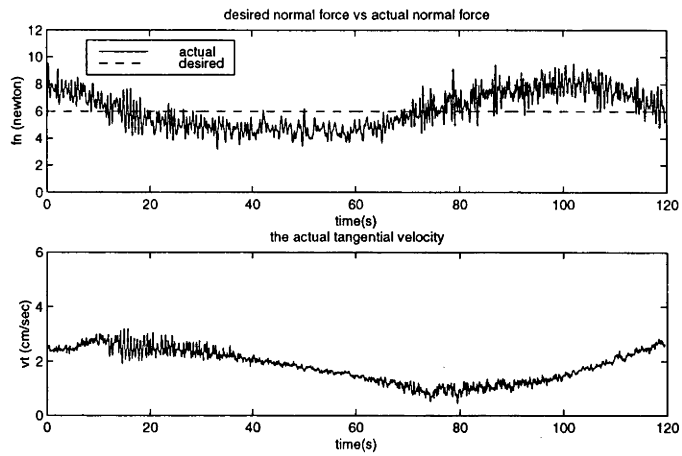


Figure 3-36: Experimental results(circular contour), $k_f = 0.5\,cm/s/N$, $\Delta t = 0.5\,s$ and $T = 0.1\,s$

are about 1262, 994 and 880 N^2 with time increment 0.5, 0.7 and 0.9 s, respectively. So the suitable time increment can be taken as about 0.8 s.

Notice that the so-called actual tangential velocity is the velocity along the estimated tangential direction. So it is only an approximation of the real tangential velocity which is almost impossible to get. Also by taking the estimated model parameters as the true values, we get the so-called normal forces. Furthermore when Δt is 0.8 s and T is 0.1 s, the results with three different gain constants are given in Figures 3-39–3-41. A small gain has a weak adjustment and a large gain has a strong adjustment but often following vibrations as shown in Figure 3-41. The performance indexes(650 sampling points) for gain 0.38, 0.52 and 0.64 cm/s/N are 936, 691 and 1069 N^2. So the gain around 0.5 cm/s/N will produce better results.

Finally, taking $k_f = 0.50\,cm/s/N$, $\Delta t = 0.81\,s$ and $T = 0.1\,s$, we use three different $\hat{\alpha}$ values, i.e. 0.4, 0.5 and 1.0 rad, in the experiments. The results are presented in Figures 3-42–3-44. So the tool can perform its contour following task with quite different model parameter estimate α. Of course, the desired normal force will not be guaranteed if the estimate $\hat{\alpha}$ is too much different from the true value.

Figure 3-45 shows the actual trajectory of the tool when gain k_f is 0.5 cm/s/N, time increment is 0.81 s and the sampling frequency is 10 Hz. The tool moves counterclockwise. After the tool completes one circle, it continues its contour following movement. So we stop it at some point. In order to have a clear understanding of the moving process, the normal force magnitude(short line segments) is added to the moving locus

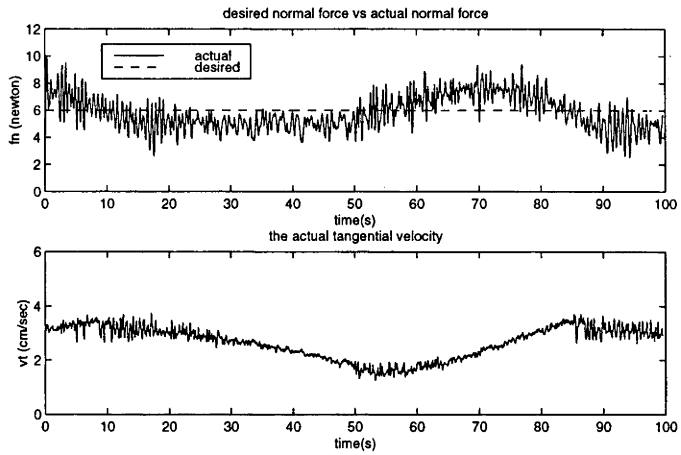


Figure 3-37: Experimental results(circular contour), $k_f = 0.5\,cm/s/N$, $\Delta t = 0.7\,s$ and $T = 0.1\,s$

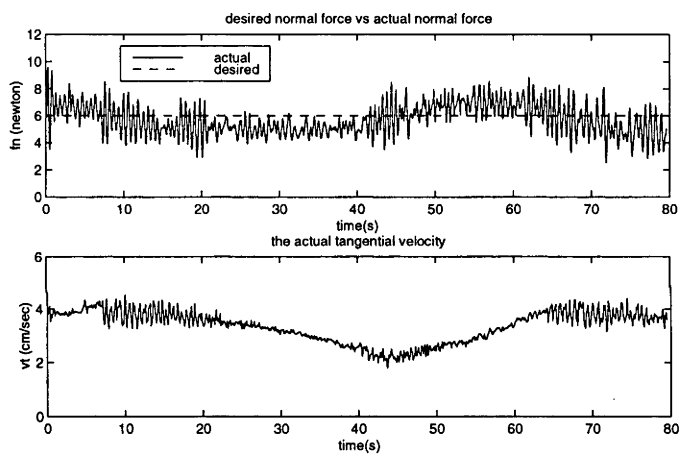


Figure 3-38: Experimental results(circular contour), $k_f = 0.5\,cm/s/N$, $\Delta t = 0.9\,s$ and $T = 0.1\,s$

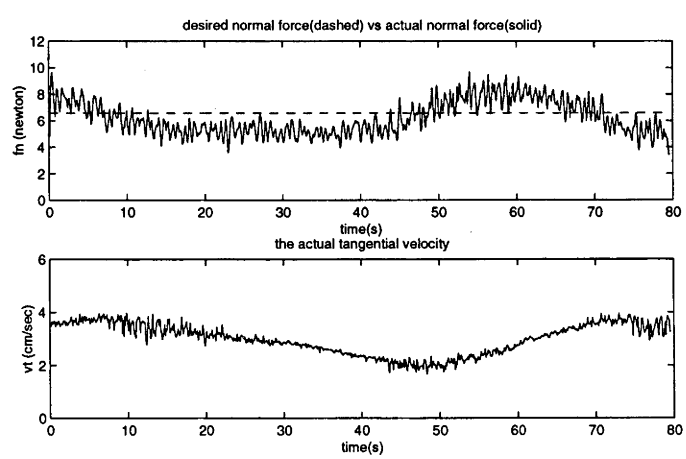


Figure 3-39: Experimental results(circular contour), $k_f = 0.38\,cm/s/N$, $\Delta t = 0.8\,s$ and $T = 0.1\,s$

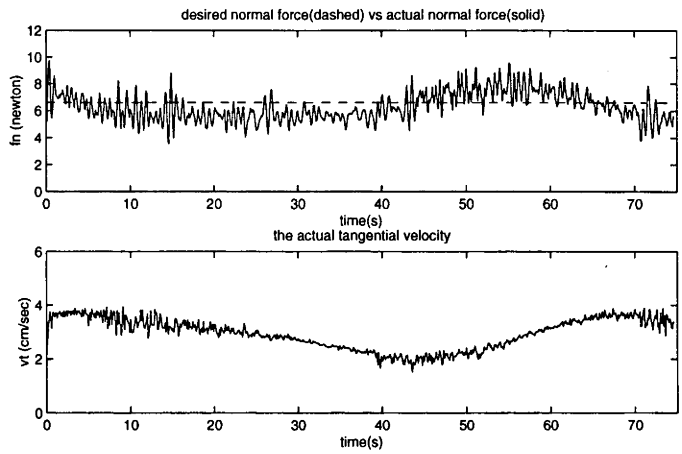


Figure 3-40: Experimental results(circular contour), $k_f = 0.52\,cm/s/N$, $\Delta t = 0.8\,s$ and $T = 0.1\,s$

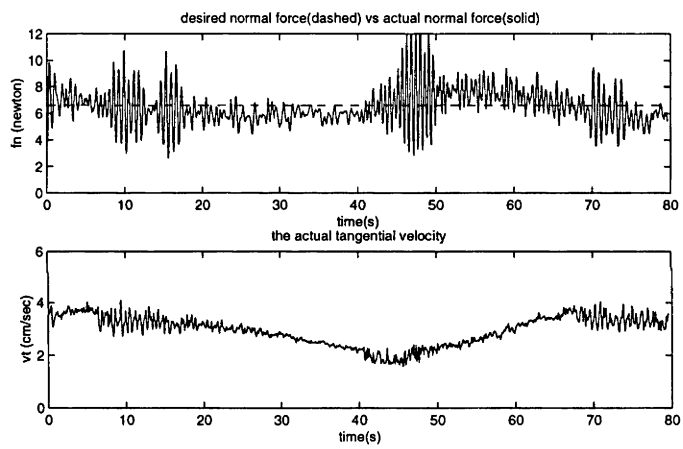


Figure 3-41: Experimental results(circular contour), $k_f = 0.64\,cm/s/N$, $\Delta t = 0.8\,s$ and $T = 0.1\,s$

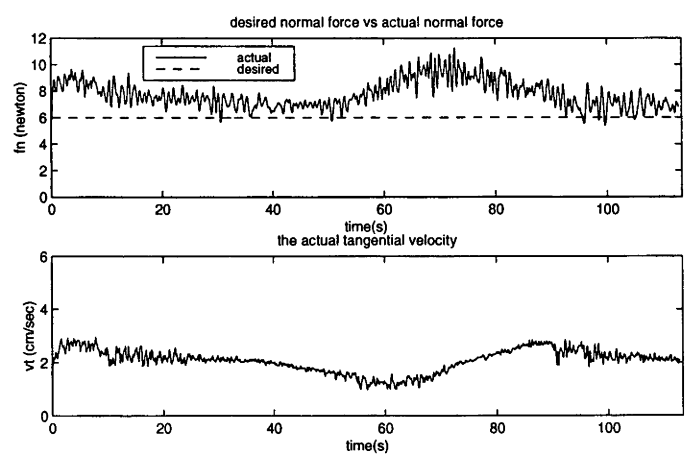


Figure 3-42: Experimental results(circular contour) with $\hat{\alpha} = 0.4\,rad$

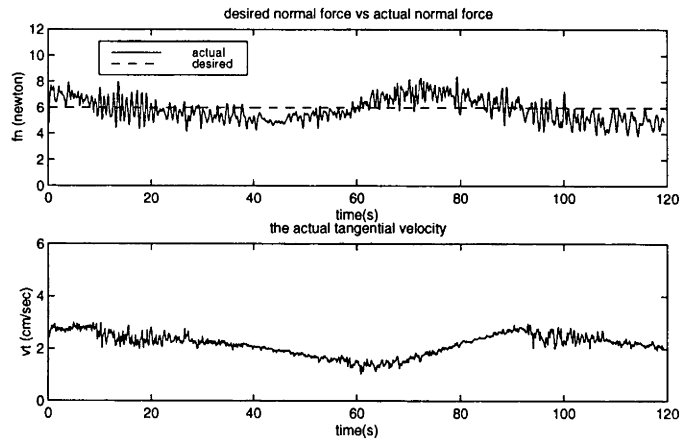


Figure 3-43: Experimental results(circular contour) with $\hat{\alpha} = 0.5\,rad$

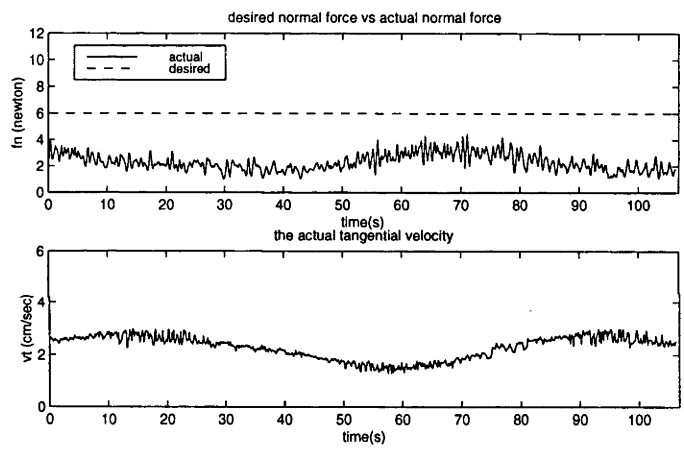


Figure 3-44: Experimental results(circular contour) with $\hat{\alpha} = 1.0\,rad$

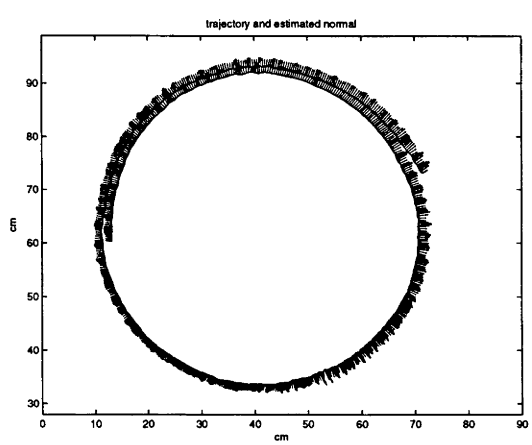


Figure 3-45: Experimental results(circular contour): $k_f = 0.5\,cm/s/N$, $\Delta t = 0.81\,s$ and $T = 0.1\,s$

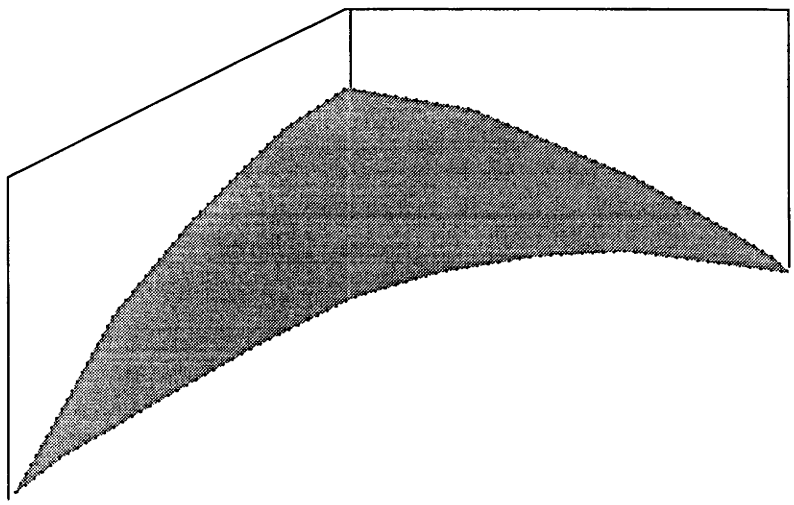


Figure 3-46: Gravel heaped around a corner

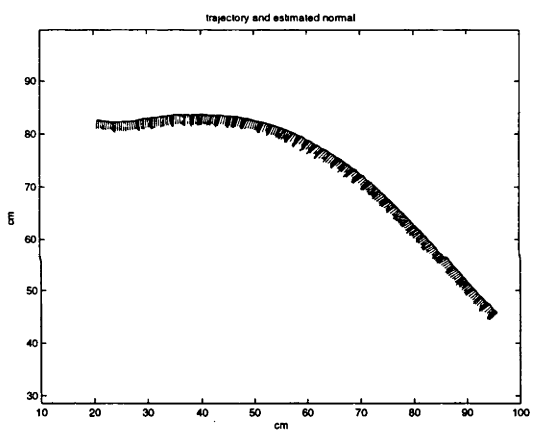


Figure 3-47: Experimental results(concave arc contour): The actual locus of the tool

of the tool. The line segment gives the estimated normal at each point. If the actual environmental structure and parameters are exactly the same as what we get, it also tell us the tool contact depth. Since the tool trajectories with other control parameters are almost the same, it is not necessary to present them.

The above experimental results illustrate that they match the simulation results and analysis in the previous section. Up to now, all of the experiments are done with the same pile shape, the circular planar contour. We now present the results with different contour shapes in the following.

First we made a corner to simulate the real environment as shown in Figure 3-46. This means that there is a pile of granular materials heaped around a wall corner.

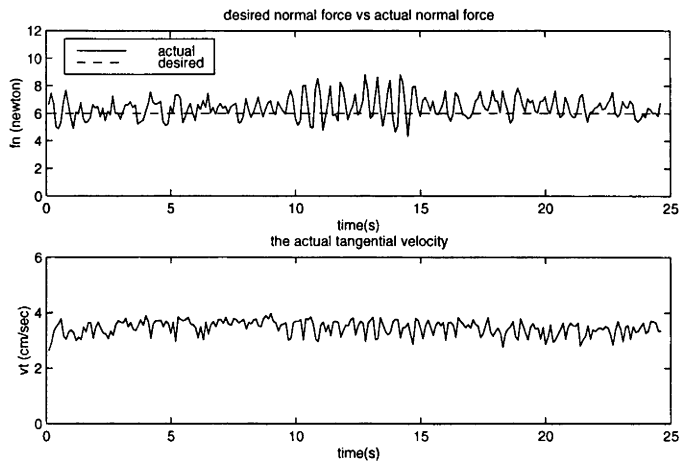


Figure 3-48: Experimental results(concave arc contour),$k_f = 0.5\,cm/s/N$, $\Delta t = 0.5\,s$ and $T = 0.1\,s$

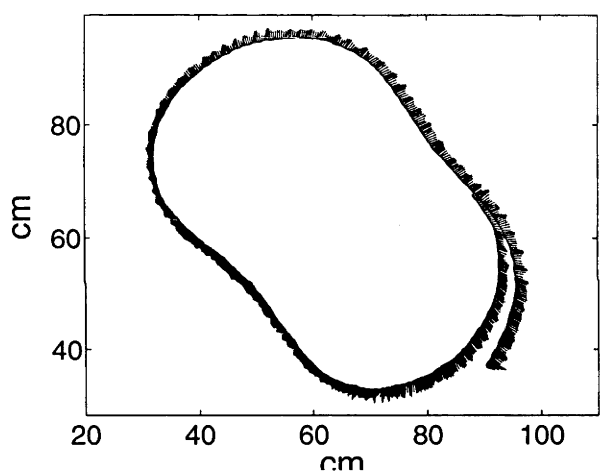


Figure 3-49: Experimental results with complex shape

We still use the former gravel; however, the planar contour is a concave arc curve. Comparatively this shape is simpler than the former circular shape. But it represents one possible actual situation. In addition, it can be used to verify the contour following control algorithm again. The model parameter α is 0.5 rad. The controller parameters are $k_f = 0.45\,rad$, $\Delta t = 0.8\,s$ and $T = 0.1\,s$. The experimental results are shown in Figure 3-47 and Figure 3-48.

Then, changing the planar contour into a little complex shape, we repeat the above experiment. This experiment is used to see if the algorithm is suitable for different and complex contour shapes. The resultant trajectory is shown in figure 3-49. The normal force and tangential velocity are shown in figure 3-50.

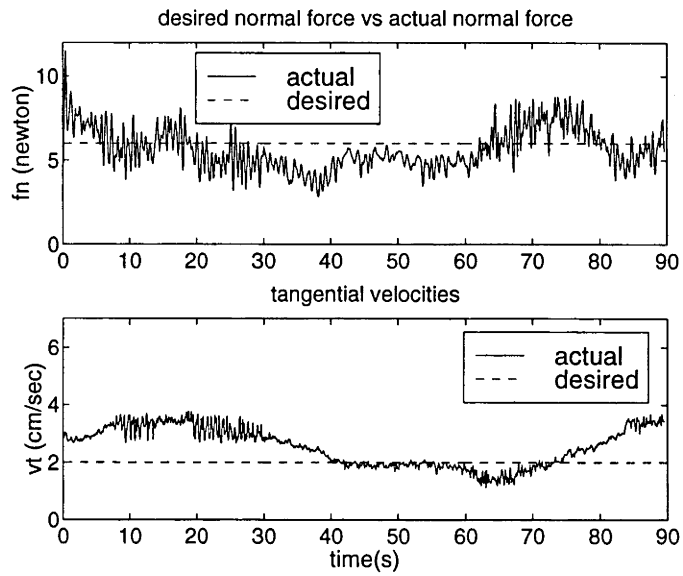


Figure 3-50: Experimental results with complex shape

From the results with the two new shapes, it is obvious that the proposed control algorithm can realize the robot tool movement over different pile planar shapes. The performance indexes(650 sampling points) for the arc and peanut-shaped shapes are 630.5 and 676 N^2, respectively. By comparison, there exists no big difference between the performance indexes of the three shapes.

All of the above tool movements except for the concave arc trajectory are in counter-clockwise direction. If the tool is required to move along the contour in clockwise(CCW) direction, the control algorithm should be modified. This can be realized by changing the directions of vector τ, $\mathbf{e_f}$ and $\mathbf{e_g}$. Accordingly, the computation of Cartesian velocities v_x and v_y is different which is illustrated in Appendix A. Figures 3-51–3-52 show the results when the tool moves along the circular contour in CCW direction. When the third contour shape is used, the tool trajectory, normal contact force and tangential velocity are illustrated in Figures 3-53–3-54.

It is worth noticing that we use the same model structure but different model parameters in the experiments. The estimated $\hat{\alpha}$ in chapter 2 is 1.26 rad which is much larger than the possible true value about 0.5 rad. The identified model parameters are mostly suitable for a tool to move straightaway across the pile. The moving direction is in between the normal and tangential of the pile surface and more near the normal. The reason is that the data obtained for parameter estimation are from this area. When

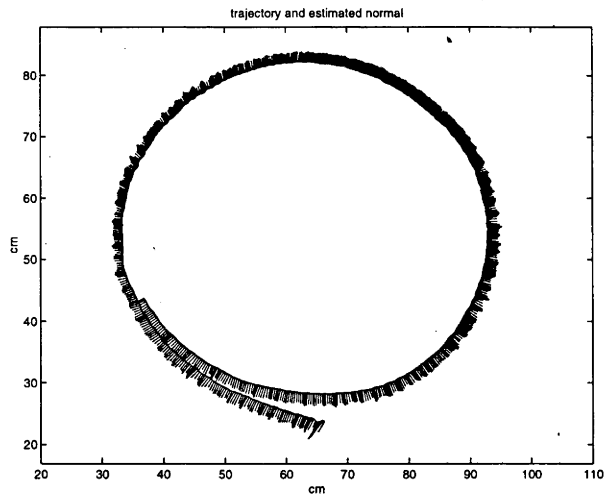


Figure 3-51: Experimental results with tool movement in CCW direction

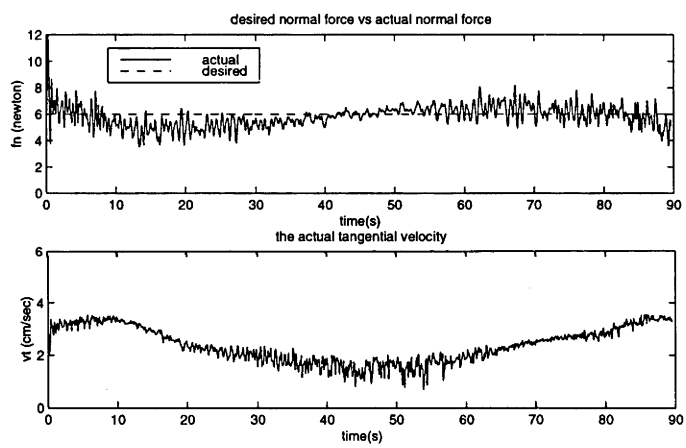


Figure 3-52: Experimental results with tool movement in CCW direction

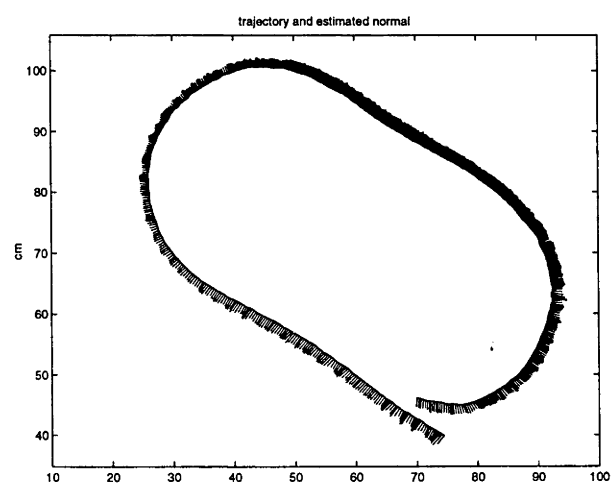


Figure 3-53: Experimental results with tool movement in CCW direction

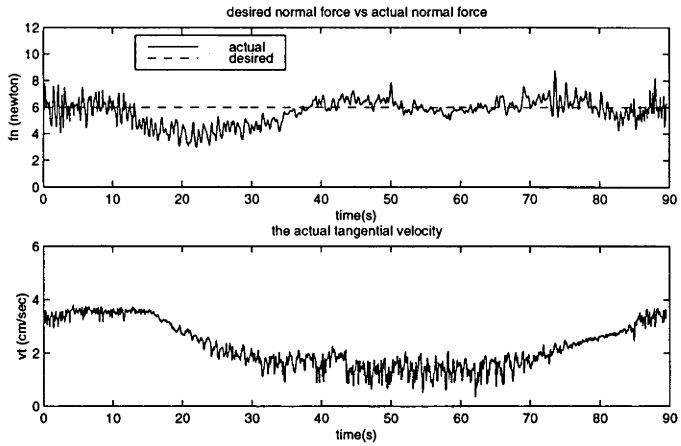


Figure 3-54: Experimental results with tool movement in CCW direction

the tool moves along the tangential, using the former identified model parameters will bring about large error. This is observed from the experimental results. In fact, we once moved the tool over a sloping plane of gravel with quite small depth. The normal force $|\mathbf{f}_n|$ is much less than $\hat{k}\,|\mathbf{f}_t|/\hat{\mu}$. On the other hand, we also once moved the tool near the normal and the sensed tangential force $|\mathbf{f}_t|$ is less than $\hat{\mu}\,|\mathbf{f}_n|/\hat{K}$. Empirically we choose $\hat{\alpha}$ 0.5 rad as the true value. From the results, this $\hat{\alpha}$ produces good force tracking results and it also gives a good estimate of the surface normal as shown in Figures 3-45, 3-47, 3-49, 3-51, 3-53.

3.10 Conclusion

Based on the model derived in the previous chapter, we develop a contour following algorithm. Force measurements as well as position information are used to determine the surface normal and tangential online. A velocity-based control system is utilised to move the robot tool over the planar contour of the pile of gravel. By slightly modifying the control algorithm, the tool can move over the pile contour in clockwise direction too. The simulation and experimental results using the SCARA robot demonstrate that the control algorithm has force tracking ability under different contour shapes. Different controller parameters are used in the experiments to determine the most suitable values. The experimental results are basically accord with the simulation results. This means that the proposed model structure and parameters are applicable. They reflect the characteristics of the actual object. Even though the SCARA robot

does not have the ability to precisely position and it has its own dynamic properties, the tracking performance is quite acceptable. The system is robust. It is not sensitive to the model parameter estimates. The controller parameters can also be chosen with large margins. Since the position control instead of the direct velocity control is used, the system cannot guarantee the desired constant tangential velocity. Nevertheless, the actual tangential velocity is satisfactory.

Chapter 4

Adaptation Mechanism for Uncertainties in Model Parameters

4.1 Introduction

In this chapter, we focus on introducing an adaptive control strategy into the contour following control algorithm. The reason for us to do so is that we cannot obtain the precise object model parameters. The object here is granular material. Even though we have derived a model to characterise the force and motion interaction between the robot tool and the gravel, the model is only approximate. In practice, materials may vary depending on moisture content etc. In addition, the model parameters are only suitable for this particular gravel.

In section 2 we give a relatively comprehensive literature review on robot adaptive control. Then an adaptive algorithm is presented in section 3. Since we explicitly estimate the model parameters on line, it is an indirect adaptive control(IAC) approach. Simulation and experimental results are given in section 4 and section 5 respectively. The results demonstrate that the proposed IAC approach is feasible.

4.2 Literature Review

The adaptive control has its roots in the mid-fifties. A comprehensive treatment of the fundamental theory of adaptive control is given in [2][15]. Adaptive robot control began in the late-seventies on account of the nonlinear, time-varying and coupled nature of robot dynamics. At the beginning, the adaptive robot control design and analysis were based on restrictive assumptions and approximations, such as linearisation of robot dynamics and decoupled joint motion. Later on, linear parameterisation of robot dynamics was introduced into adaptive robotic control. Using the system equivalent principle, a set of parameters is chosen such that robot dynamics depend linearly upon these parameters. More recently, adaptation mechanism has been used in robot control to solve various problems. For example, besides the manipulator dynamics, the load change and the uncertainties in the environmental parameters need to be considered when the robot end-effector interacts with the environment or performs its tasks.

Normally, there are three categories of adaptive strategies for robot force control: indirect, direct, and composite methods. In the indirect adaptive control(IAC) [11] [24] [28] [38], there is an explicit parameter estimation of the controlled robot system. The adaptation algorithm is obtained by using the predictive error related to the estimation. This adaptive method does not consider the actual tracking errors. In the direct adaptive control(DAC) the adaptation algorithm is adjusted by the trajectory tracking errors [8] [30]. The self-adjusting mechanism in the adaptation algorithm tries to make the tracking errors converge to zero. Compared to the indirect adaptive method, the online computation task in direct adaptive control is much larger. The composite adaptive control(CAC) strategy combines the indirect and direct approaches. Its adaptation algorithm makes use of both tracking error and prediction error [39] [55].

We introduce some single rigid robot adaptive force control methods in three aspects: adaptive impedance control, adaptive hybrid control, and adaptive force-position control.

4.2.1 Adaptive impedance control

Recognising that it is impossible to control contact force and position of robot end-effector simultaneously when it is in constrained motion, Hogan [18] [19] put forward

the impedance control concept. The main idea in impedance control is that the position and force can be controlled by regulating the mechanical impedance of manipulators. The impedance control has been implemented in many ways.

However, conventional impedance control requires the precise knowledge of manipulator dynamics and the perfect measurement of external force. Generally, there are some uncertainties in parameters of manipulator dynamic model because of the modelling complexity and the unpredictable payload changes. Besides, the measured contact force are often mixed with noise. So these factors will affect the quality of impedance control implementation. In order to overcome the disadvantages, Lu and Meng [26] proposed an adaptive impedance control strategy. First they introduced the concept of target-impedance reference trajectory(TIRT), which characterises the desired dynamic relation between robot end-effector and its environment. Then an adaptive mechanism is injected into Hogan's impedance control. And a new Lyyapunov approach is utilised to properly compensate the bounded measurement noise in force sensor.

As for indirect adaptive control, there are three major disadvantages: the computation is intensive when the manipulator has more than two or three joints; the precise knowledge of manipulator model parameters; lack of robustness. The direct adaptive impedance control method proposed by Colbaugh, Seraji and Glass [8] tries to cope with those problems. It has three major parts: a simple 'filter' is used to characterise the desired dynamic relations of end-effector position error and interaction force; an adaptive controller which provides the control input required for that dynamic relationship; a mapping algorithm. The controller does not need the knowledge of manipulator model parameters. It does not need to compute the inverse kinematic transformation too. In addition, it makes use of the redundancy effectively. Similar adaptive impedance control strategies were proposed in [34]

In the robotics literature, there are many control strategies to deal with position tracking and force regulation when a robot manipulator interacts with compliant and infinitely rigid environments. But very few consider the transient behaviour of a closed loop system between contact and non-contact. Mills and Lokhorst [29] gave a complete analysis for contact and non-contact transient phases. And the proposed algorithm guarantees transient stability. On the basis of this method, Wit and Brogliato proposed another direct adaptive impedance control [49]. Its most important merit is that the

overall system stability is considered without a priori knowledge of the environmental stiffness constant.

The Adaptive control scheme [44] proposed by Tzafestas *et al* consists of two stages of adaptation and control. The first one carries out an on-line estimation of the robot inertial parameters. The second one compensates for the uncertainties on the characteristics of the ground(position and stiffness) when a robot leg is in constrained and unconstrained motion. A set of simulation results were presented to illustrate the capabilities of the proposed scheme.

4.2.2 Adaptive hybrid control

In order to apply robot manipulators to a wide variety of tasks, it is necessary to control not only the position of a robot end-effector, but also the force exerted on the environment by robot end-effector. The hybrid control method has been proposed by Rabert and Craig [35]. The aim of hybrid control is to split up simultaneous control of robot-effector motion and contact force into two separate and decoupled subproblems. In some directions, end-effector position is controlled and the force is controlled in other directions. Hybrid control has many implementations.

When there are uncertainties in robot dynamic parameters, in environment dynamics, or in both, some adaptation mechanism should be introduced to hybrid control for high-quality control. Some important adaptive hybrid control strategies are proposed by Jean and Fu [21], Slotine and Li [40], and Lozano and Brogliato [25]. The general strategies of these authors are that both a hybrid control law and an adaptive law were designed for a robot with unknown dynamic parameters. So a robot end-effector can accurately track the desired trajectory in the unconstrained directions and exert the desired force in the constrained directions. However, these proposed controllers are only designed for adaptation for manipulators, not for the environment dynamics.

More recently, J. Wu *et al* [50] proposed an implementation of adaptive hybrid control. The considered environment is uncertain flexible objects. Since a flexible object is usually a distributed parameter system, the object dynamics as seen from the robot changes when the robot moves. J. Wu *et al* approximated the object's distributed parameter model into a lumped 'position state-varying' model. The robot control space is decomposed into a position control subspace and an object torque control subspace.

Then an optimal state feedback is used for the position control loop and the resultant torque of the object is used to control the contact force. But this method assumes that there are no uncertainties in the robot's dynamics.

One important adaptive model-based hybrid control was proposed by L.L. Whitcomb *et al*[48] and S. Arimoto *et al* [1]. This class of model-based adaptive force control algorithms is applied to a robot tool's constrained motion over a smooth rigid environment or workpiece. They used a sliding-mode to provide asymptotically precise tracking of both end-effector position and contact force. The introduced adaptation law can adaptively compensate for unknown plant parameters such as link and payload inertia, and joint friction. The authors made use of many experimental results to illustrate the advantages of the proposed algorithms.

Many schemes for robot motion control design robot controllers at torque input level and actuator dynamics are excluded. But when robot moves at high velocity with highly varying load, the actuator dynamics constitute an important part of the entire robot dynamics. C. Su and Y. Steponenko [43] proposed a method to deal with the uncertainties both in the manipulator dynamics and in the actuator dynamics. A hybrid adaptive/robust control law was proposed for an n-link manipulator which includes the effects of actuator dynamics.

4.2.3 Adaptive force-position control

In this subsection, we present some special adaptive control strategies for robot constrained motion. These methods are different from those we discussed in the above subsections.

For a dynamic environment, Yao and Tomizuka [51] [52] proposed an adaptive control strategy. The proposed strategy has a number of properties: a new transformed constrained dynamic model is used and it is suitable for controller design and valid for a friction contact surface; both motion and force tracking errors are used for adaptation law; the control law can guarantee asymptotical motion and force tracking without persistent excitation condition; uncertainties both in robot dynamics and in surface friction factor are considered; the suggested controller has robustness to the bounded measurement noises in the force and velocity sensors as well as bounded disturbances. But the work was done only theoretically.

The adaptive control strategy proposed by Ekalo and Vukobratovic [12] [13] deals with dynamic environment described with a nonlinear differential equation. The proposed adaptive law is to guarantee simultaneous stabilisation of motion and interaction force when inadequate description of the robot dynamics exists, or when robot parameters change with time in an unknown way. This method is aimed at direct realization of a desired robot motion and desired contact force which, as a pair of time functions, satisfy the general nonlinear second-order differential equation of the environment dynamic model.

On the basis of parallel force/position control [6] [42], B. Siciliano and L. Villani [37] proposed an adaptive method for robot compliant motion. This control law is designed in the task space and contains a nonlinear model-based term and a linear compensation action. The position error, the velocity error and the integral of force error are all used for the control algorithm. Force and position variables are used along each task space direction without any switch mechanism. The conflict between force and position along the constrained task directions is resolved with the dominance of the force-control loop over the position-control loop. The scheme is made adaptive with respect to the dynamic model parameters of the manipulators.

A new parallel control scheme proposed by Chiaverini *et sl* [7] performs tracking of the contact force along the constrained task direction as well as tracking of the end-effector position along the unconstrained task direction. A model-based controller of inverse dynamics type with force feedforward is designed. Adaptation to unknown stiffness is produced with an estimate update law which is driven by a properly filtered version of the force error. The proposed scheme is experimentally tested.

Now let us have a look at some strategies in explicit adaptive force control. There are two broad types of explicit adaptive force controls. In the first method, the manipulator joint actuators can be directly controlled so the joint torques are used as the control input [28] [34]. The second approach is motivated by the fact that typically the industrial robots are equipped with position controllers. The position control is used as inner loop control and the outer loop is the force control. The inner loop and the outer loop are combined to become a single system. The control input is a position command and the adaptation algorithm is introduced in the outer loop [47]. This method ensures global uniform boundedness of all signals. It also makes posit-on/force

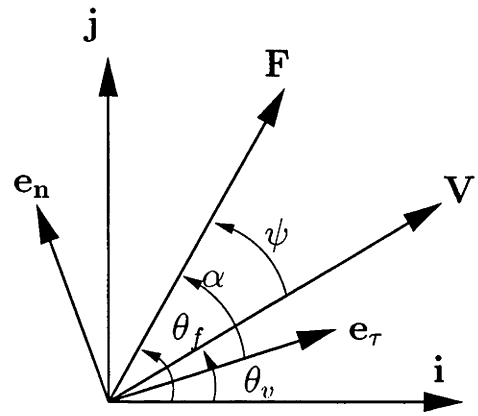


Figure 4-1: Force and velocity under polar coordinate

errors converge to zero. Two adaptive algorithms are applied for PD position-control and PID position-control manipulators respectively. In addition, a robust mechanism is applied for compliant motion of PD position-control manipulators.

Adaptive control is a broad discipline. Many new concepts and ideas are introduced into adaptive control. For example, introducing fuzzy logic and neural network yields fuzzy logic adaptive control(FLAC) and neural network adaptive control(NNAC) in robot control. They are recently quite active research fields.

4.3 Adaptive Algorithm

In this section we propose an IAC mechanism. The model parameters are explicitly estimated on line. The parameters are then used for the controller.

4.3.1 Description of System Input and Output

In this subsection we derive a new relationship between system input and output. The objective is to derive equations that are linear in the unknown parameters. Then we can conveniently introduce the adaptation mechanism.

Figure 4-1 shows the relationship between tool velocity $\mathbf{V}$ and contact force $\mathbf{F}$. Coordinates $(\mathbf{i}, \mathbf{j})$ are related to reference frame(Cartesian space). $\mathbf{e_n}$ and $\mathbf{e}_\tau$ are still the unit normal and tangential vectors. Obviously, the measured parameters are $(|\mathbf{F}|, \theta_f)$ and the imposed parameters(control inputs) are $(|\mathbf{V}|, \theta_v)$. All of these quantities are known. From the former model 2.1 and 2.2, we have

$$\begin{cases} F_\tau = \frac{\mu}{k} F_n \\ \dot{F}_n = k V_n \end{cases} \tag{4.1}$$

where F_n and F_τ are normal and tangential force magnitudes, $\dot{F}_n$ is the normal force differential, and V_n is normal tool velocity. From Figure 4-1, we obtain

$$\begin{cases} F_\tau = F \cos\alpha \\ F_n = F \sin\alpha \\ V_n = V \sin(\alpha - \psi) \\ \dot{F}_n = \dot{F} \sin\alpha \end{cases}$$

Notice that we assume model parameters are time invariant. Furthermore

$$\begin{aligned} \dot{F} \sin\alpha &= k V \sin(\alpha - \psi) \\ &= k V(\sin\alpha \cos\psi - \cos\alpha \sin\psi) \end{aligned}$$

Dividing the two sides of the above equation by $\cos\alpha$ and using $\tan\alpha = k/\mu$ yields

$$\dot{F} = V(k \cos\psi - \mu \sin\psi) \tag{4.2}$$

When the sampling time interval is small enough, we can take

$$\dot{F} = \frac{\Delta F}{\Delta t}$$

$$V = \frac{\Delta P}{\Delta t}$$

where ΔF, ΔP and Δt are force, position and time increments, respectively. Then equation (4.2) can be written as

$$\Delta F = \Delta P(k \cos\psi - \mu \sin\psi) \tag{4.3}$$

where $\psi = \theta_f - \theta_v$

From Figure 3-7 we have

$$\theta_f = \theta_1 + \theta_2 + atan2(f_y, f_x)$$

$$\theta_v = atan2(\Delta y, \Delta x)$$

where Δx and Δy are Cartesian position increments.

The unknown parameters appear linearly in equation (4.3). So equation (4.3) makes on-line estimation of parameter k and μ possible and easier.

4.3.2 Indirect Adaptation Method

In this subsection we combine a parameter estimation algorithm and a control law to give an IAC algorithm. Firstly, we use recursive least squares(RLS) method for model parameter estimation. Because of the rich dynamic properties of the granular material, the model parameters will not be constant even for a particular pile. So a forgetting factor is included to cope with the slowly time-varying property.

In order to apply the identification algorithm, we rewrite equation (4.3) as

$$\Delta F = kmu^T \varphi(t) + \epsilon(t) \tag{4.4}$$

where

$$\Delta F = F(t+1) - F(t)$$

$$kmu = (k \quad \mu)^T$$

$$\varphi^T = (\Delta P cos\psi \quad -\Delta P sin\psi)$$

and ϵ is white noise. This system can be treated as two-input one-output system. Two parameters, k and μ need to be identified. Notice that input R is related to output F since θ_f is the angle of current force vector. It seems that a feedback exists between the input and the output. This means that the identification is operating in closed loop. For simplicity, we directly apply an open loop identification method without consideration of the feedback as long as it works [41]. The RLS with a forgetting factor algorithm is as

$$\begin{cases} P(t) = \{P(t-1) - P(t-1)\varphi(t)\varphi^T(t)P(t-1)/[\lambda + \varphi^T(t)P(t-1)\varphi(t)]\}/\lambda \\ K(t) = P(t)\varphi(t) \\ \varepsilon(t) = \Delta F(t) - \varphi^T \hat{kmu}(t-1) \\ \hat{kmu}(t) = \hat{kmu}(t-1) + K(t)\varepsilon(t) \end{cases} \tag{4.5}$$

where $P(t)$ is a diagonal matrix, λ is a forgetting factor. The identification process can be expressed as follows. First some initial values are given. For example, P(0) is a diagonal matrix whose diagonal components should be large numbers such as 10^5. $\hat{kmu}(0)$ can be given the initial estimated values. The number of iterations is then given. Finally the algorithm is executed recursively.

Choosing forgetting factor $\lambda(t)$ is not trivial. There are two kind of forgetting factors: constant forgetting factor(CFF) and time-varying forgetting factor(TVFF). CFF is a positive constant less than one. It is applicable to time-invariant system and slowing time-varying systems. When a time-varying property is obvious and particularly if the parameters will change abruptly, TVFF is preferable. It will be updated on-line. For example, when the parameters are invariant, λ will approach 1 quickly. However, if the parameters change abruptly, λ will decrease immediately such that the system can quickly follow the parameter change. Comparatively, a system with TVFF keeps track of the parameter variation more quickly. It also has higher identification precision. Algorithms with small forgetting factors become sensitive. So the parameter estimates approach the true values rapidly, but it will more easily bring about large oscillations , especially in noisy situations. If CFF is used, it should be about 0.9 empirically. For simplicity and because possible slowly-time-varying property are expected, we use a constant forgetting factor(CFF).

If we want to move the tool in the CCW direction, we use equation (3.15)-(3.17) to determine the next desired position of the tool.

4.4 Simulation Results

In this section, we present the results of two different simulations. The supposed planar contour is still an ellipse.

4.4.1 Simulation Using Virtual Ideal Robot

It is first supposed that an ideal robot is used to perform point-to-point movement over the simulated pile of gravel. We also use velocity-based position control approach as shown in Figure 3-6. Note that according to the parameter estimation theory, if the system input signals are sufficiently exciting and the estimated model is appropriate,

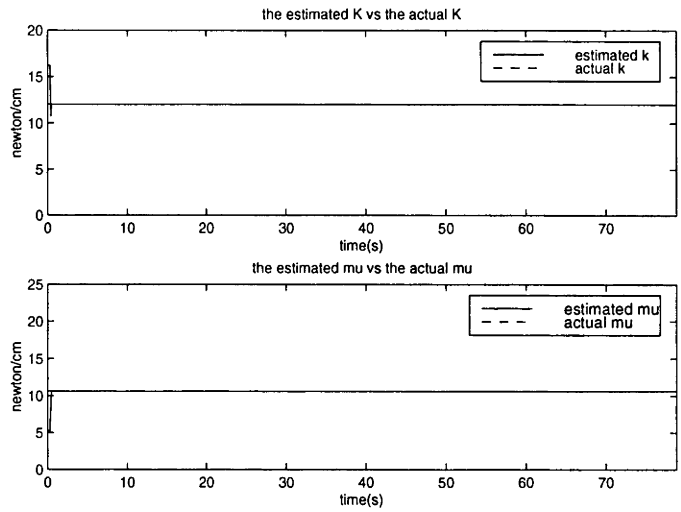


Figure 4-2: Simulation results: Parameter estimates $\hat{k}$ and $\hat{\mu}$

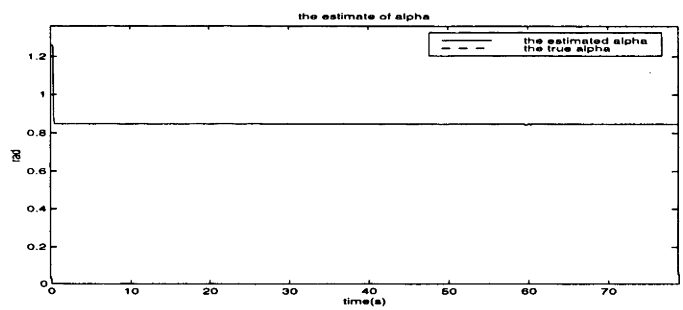


Figure 4-3: Simulation results: Angle estimate $\hat{\alpha}$

the estimates will converge to the true values provided that the closed-loop system is stable. In order to satisfy the requirement of sufficient excitation, we add a random noise to the desired constant normal force. The noise level is between -0.5 and 0.5 N. Otherwise, the system is non-identifiable without sufficient excitation.

Figure 4-2 shows the estimated parameters $\hat{k}$ and $\hat{\mu}$. The constant forgetting factor(CFF) is 0.98 and k_f is 0.2. The time increment is 0.1 s. The velocity constant V_c is 2 cm/s which is the same for this whole thesis. The initial parameter values are: $k(0) = 16.2$, $\mu(0) = 5.2$. The true values are: $k = 12$, $\mu = 10.6$. It is clear that the parameter estimations quickly approach the actual values. Figure 4-4 shows the normal contact force and the tangential velocity. The actual tangential velocity soon reaches the desired value. Since a noise is added to the desired constant normal force, the resultant desired normal force and the actual normal force are both noisy. But the actual normal force is around the desired constant normal force. The results without

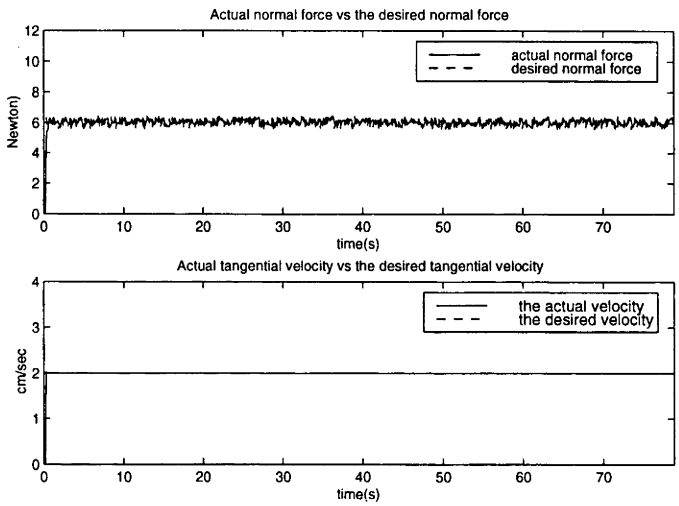


Figure 4-4: Simulation results: Normal force and tangential velocity

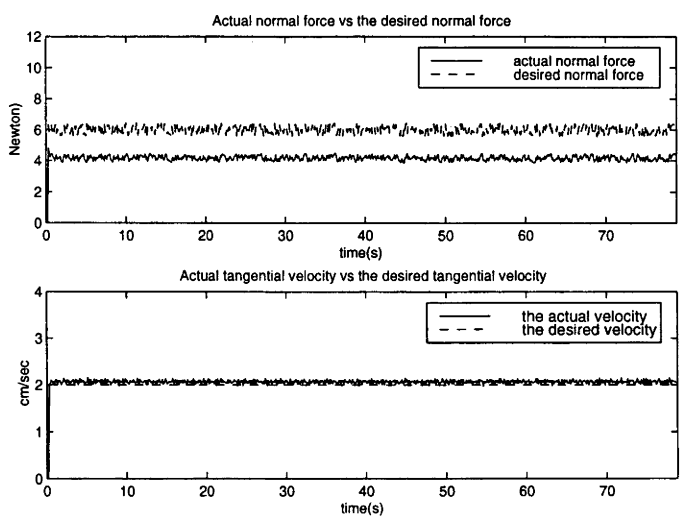


Figure 4-5: Simulation results without adaptation: Normal force and tangential velocity

adaptation mechanism are shown in Figure 4-5.

By comparison, it is clear that the results with adaptation mechanism are much better than those without adaptation mechanism. When the estimated $\hat{\alpha}$ is much larger than the true value, the actual normal force will be much less than the desired value. Also there is a relatively large difference between the actual and desired tangential velocities when the adaptation mechanism is not adopted. These similar results have already been shown in the previous chapter. So when high control performance is required and object parameters cannot be obtained in advance and/or off-line, adaptive control is preferable. Note that the identification is under closed loop. Nevertheless, we can get the approximate estimation of the model parameters.

4.4.2 Hybrid Simulation Using Real Robot

In this subsection, we present the hybrid simulation results. As mentioned in the previous chapter, the hybrid simulation is a hybrid combination of simulation and experiment. The robot is real and motion data is experimental. The controller parameters are $k_f = 0.88\,cm/s/N$, $\Delta t = 0.92\,s$ and $T = 0.1\,s$. We use three different forgetting factors to see how the forgetting factor affects the system performance.

Figures 4-6–4-8 give the results of the parameter estimation. The results of the corresponding angle estimation $\hat{\alpha}$ are shown in Figures 4-9–4-11. Figures 4-12–4-14 show the normal forces and tangential velocities. Figure 4-15 gives the results when the adaptation mechanism is not used with $\hat{\alpha} = 0.9\,rad$. Note that random noise is added to the desired normal force.

The hybrid simulation results demonstrate that the IAC algorithm can produce quick estimation of the model parameters. There is error in the estimation but the error is very small. It does not affect the system performance largely. Using the integral square-error criterion yields that the suitable forgetting factor is around between 0.97 and 0.99. Since the model parameters can be obtained quickly online, the system performance will be improved largely compared to the case where adaptation is not used and there exist relatively large errors in the estimated model parameters.

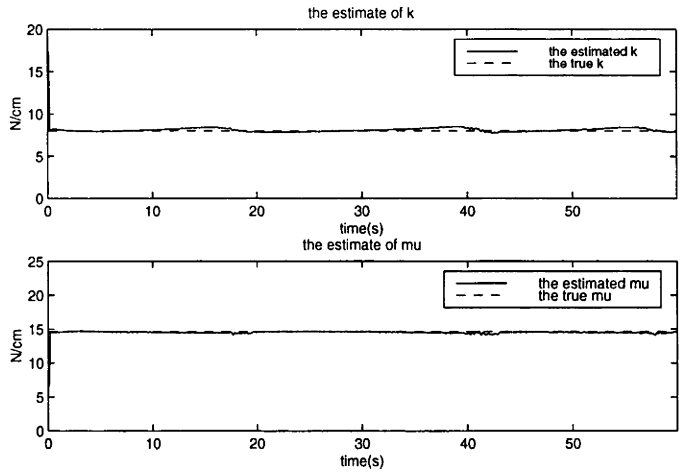


Figure 4-6: Hybrid simulation results: parameter estimation with $\lambda = 0.90$

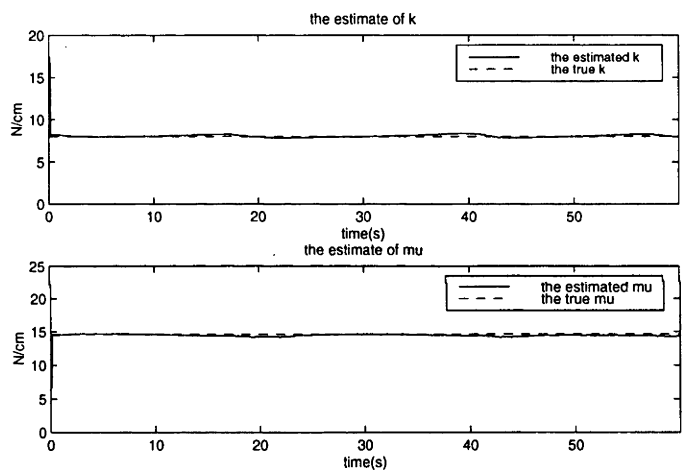


Figure 4-7: Hybrid simulation results: parameter estimation with $\lambda = 0.97$

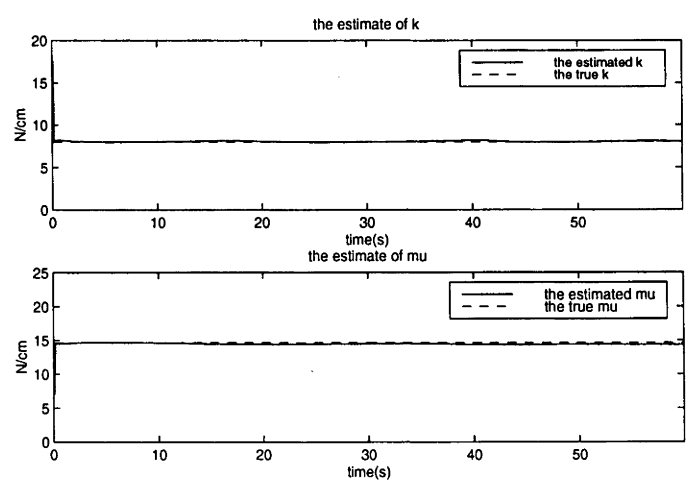


Figure 4-8: Hybrid simulation results: parameter estimation with $\lambda = 0.99$

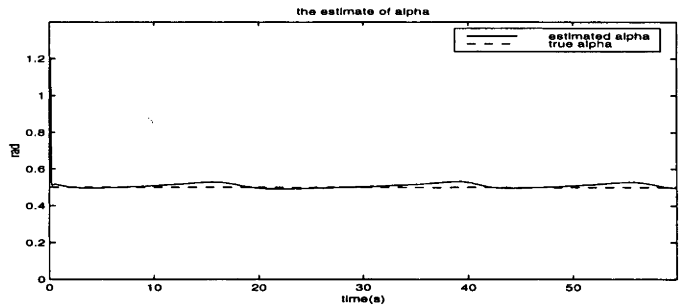


Figure 4-9: Hybrid simulation results: angle estimation with $\lambda = 0.90$

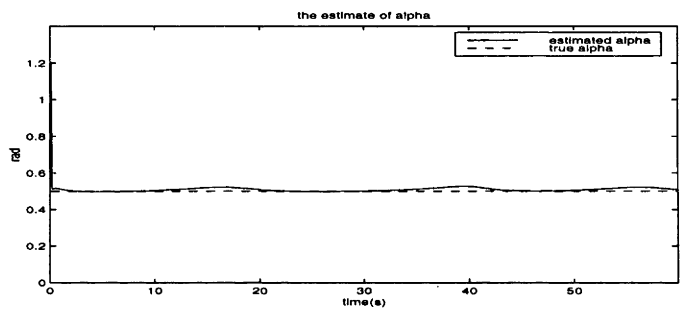


Figure 4-10: Hybrid simulation results: angle estimation with $\lambda = 0.97$

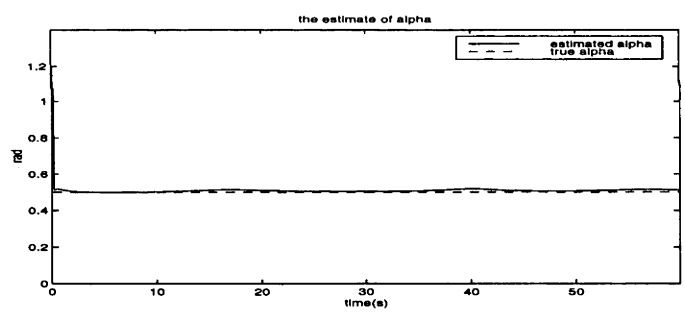


Figure 4-11: Hybrid simulation results: angle estimation with $\lambda = 0.99$

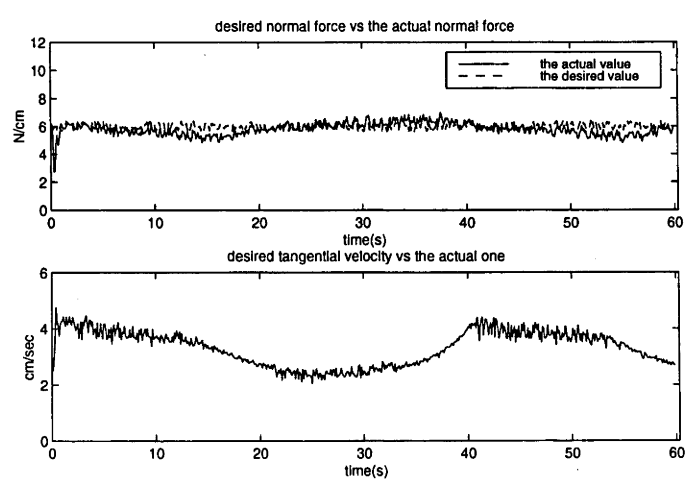


Figure 4-12: Hybrid simulation results: force and velocity with $\lambda = 0.90$

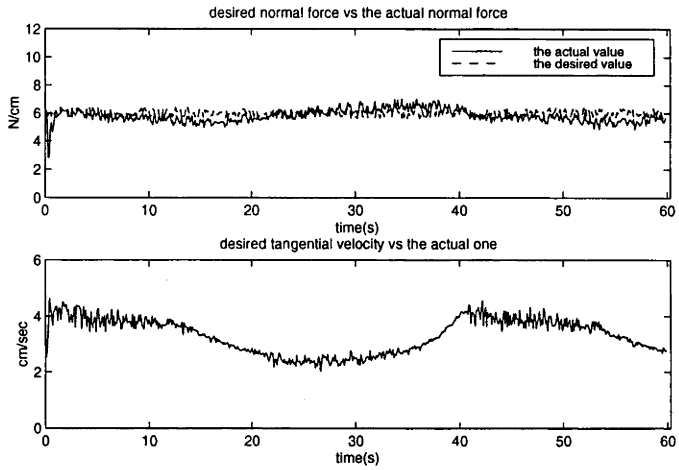


Figure 4-13: Hybrid simulation results: force and velocity with $\lambda = 0.97$

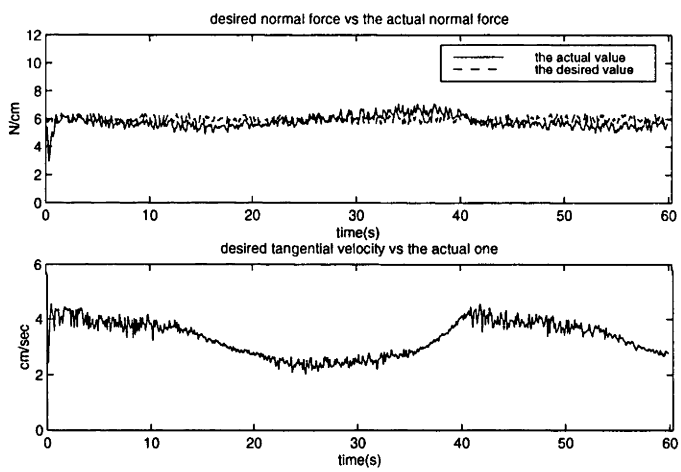


Figure 4-14: Hybrid simulation results: force and velocity with $\lambda = 0.99$

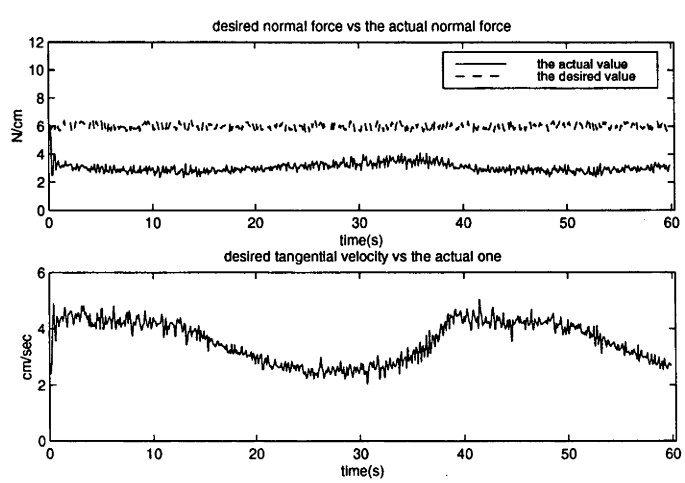


Figure 4-15: Hybrid simulation results: force and velocity without using adaptation

4.5 Experimental Results

In this section we present the experimental results related to the IAC method.

Figures 4-16–4-19 show the parameter estimation results when four different forgetting factors are used. Figures 4-20–4-23 give the corresponding estimates of the angle α which is the angle between the force vector and the surface tangential. The actual normal forces and tangential velocities related to these four angles are illustrated in Figures 4-24–4-27. Figure 4-28 shows the normal force and the tangential velocity without using adaptation mechanism with $\hat{\alpha} = 0.9\,rad$. Note that the results shown in Figures 4-24–4-28 are obtained on the assumption that the true α is 0.5 rad. The initial estimate of α is 0.9 rad. The controller parameters are $k_f = 0.46\,cm/s/N$, $\Delta t = 0.82\,s$ and $T = 0.1\,s$. In addition, $\hat{\alpha}$ may change largely, especially when the first estimate of α is produced online. This will bring about large oscillation in the force. One simple approach is to limit the increment of α each time for the controller parameter computation. Applying the error performance criterion, the performance indexes for the four forgetting factors 0.93 through to 0.99 are 1279, 1150, 1086 and 905 N^2, respectively. It shows that larger forgetting factor yields better results.

We can see that although the estimated parameters $\hat{k}$ and $\hat{\mu}$ change with relatively large magnitude, the estimate $\hat{\alpha}$ changes little. It is around between 0.4 and 0.6 rad. It is clear that the forgetting factor λ affects the system performance. With the consideration of both parameter estimation and control performance, the suitable forgetting factor is about 0.97.

In addition, we added a random noise into the desired normal force. The noise magnitude is between -0.4 to 0.4 N. The forgetting factor is chosen to be 0.97. The controller parameters are the same. The results are shown in Figures 4-29–4-31. It is clear that the results with noise are very similar to those without noise. This means that the actual system input is basically sufficiently exciting. It is not necessary to add extra noise.

The experimental results demonstrate that it is feasible to use the RLS(recursive least squares) method to estimate model parameters online. Even though the model parameters are not constant, the IAC approach can at least yield their approximate values. The performance with adaptation is better than that without adaptation as shown in Figures 4-24–4-28. The results also illustrate that the actual α is around 0.5

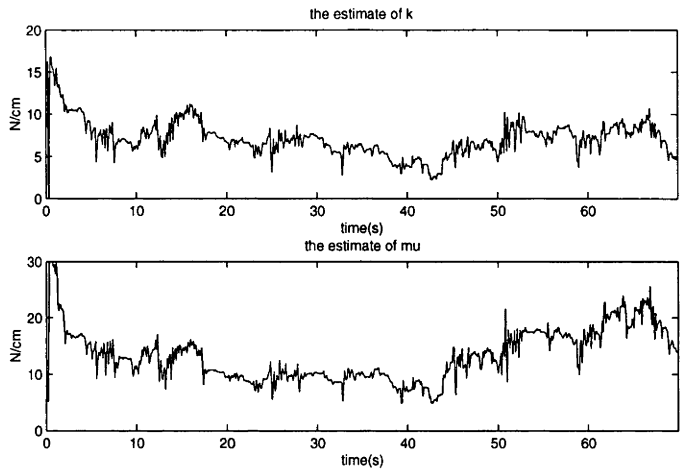


Figure 4-16: Experimental results: parameter estimation with $\lambda = 0.93$

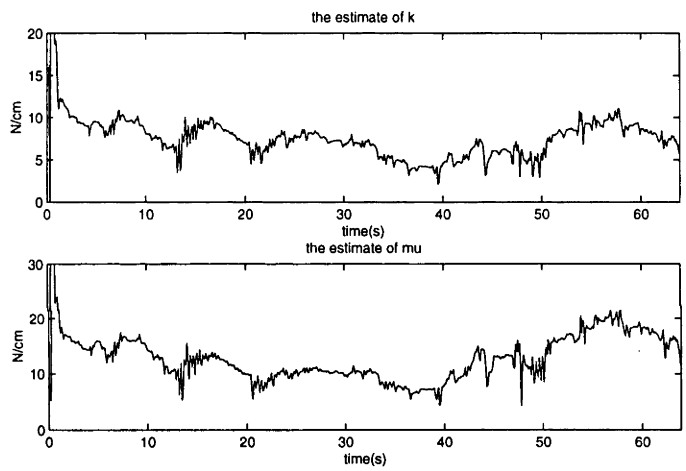


Figure 4-17: Experimental results: parameter estimation with $\lambda = 0.95$

rad when the tool moves on the contour with shallow depth. In addition, they also demonstrate that the surface normal estimate in chapter 3 is acceptable.

4.6 Conclusion

Considering the possible need for an adaptation mechanism to deal with the uncertainties in the gravel parameters, we derive a new description of the system input and output. The combination of on-line RLS parameter estimation and feedback control brings about a simple IAC method. The experimental results demonstrate that the interaction model parameters are not constant. But the IAC approach can at least produce their approximate values. The simulation and the experimental results also

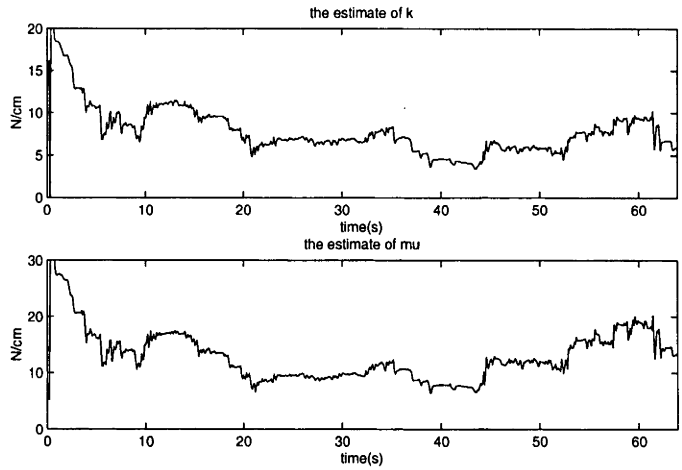


Figure 4-18: Experimental results: parameter estimation with $\lambda = 0.97$

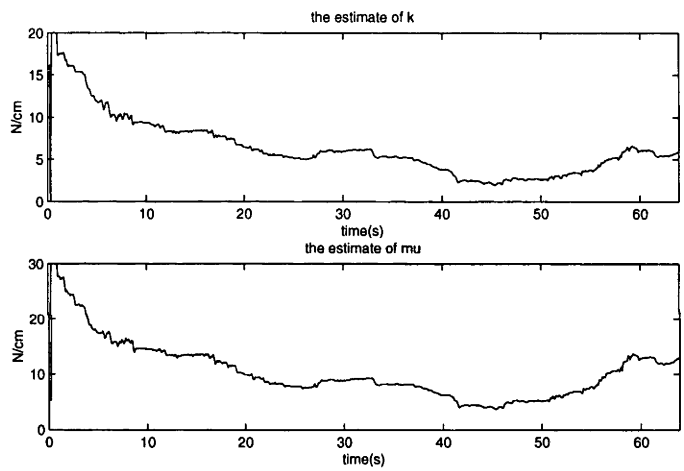


Figure 4-19: Experimental results: parameter estimation with $\lambda = 0.99$

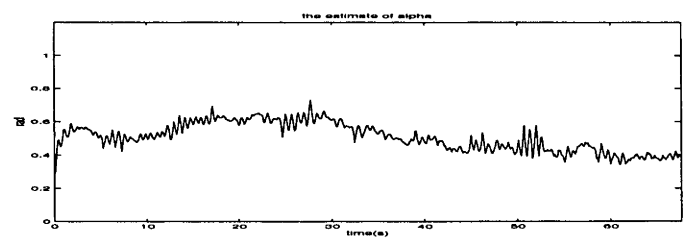


Figure 4-20: Experimental results: Angle estimation with $\lambda = 0.93$

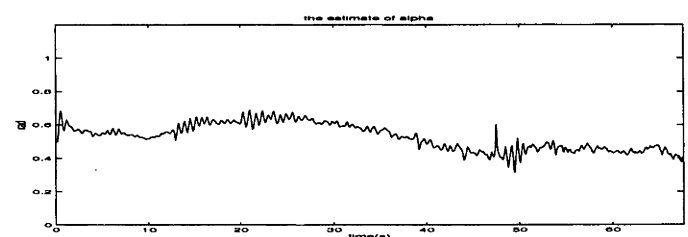


Figure 4-21: Experimental results: Angle estimation with $\lambda = 0.95$

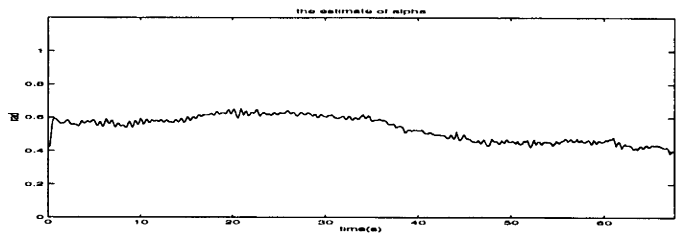


Figure 4-22: Experimental results: Angle estimation with $\lambda = 0.97$

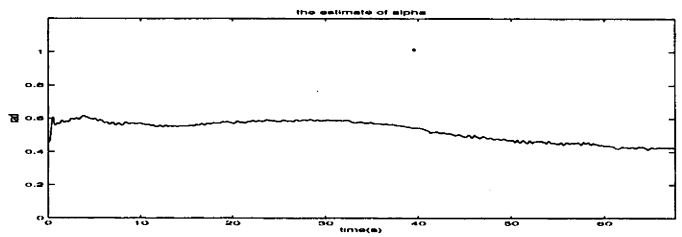


Figure 4-23: Experimental results: Angle estimation with $\lambda = 0.99$

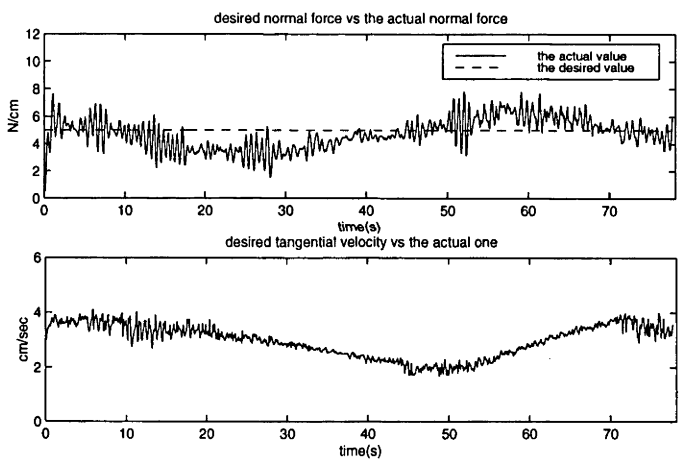


Figure 4-24: Experimental results: Normal force and tangential velocity with $\lambda = 0.93$

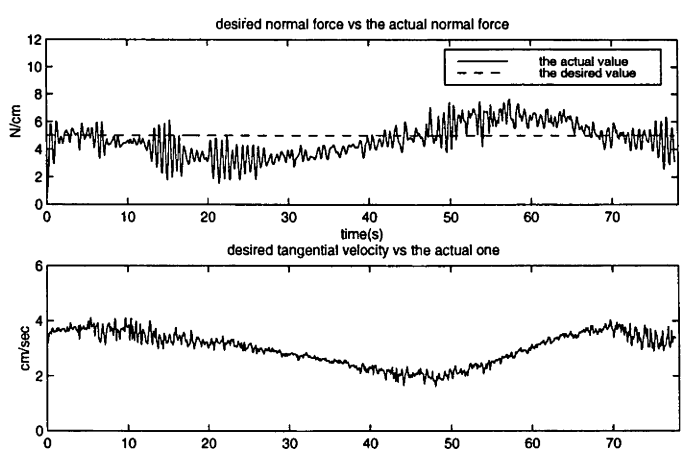


Figure 4-25: Experimental results: Normal force and tangential velocity with $\lambda = 0.95$

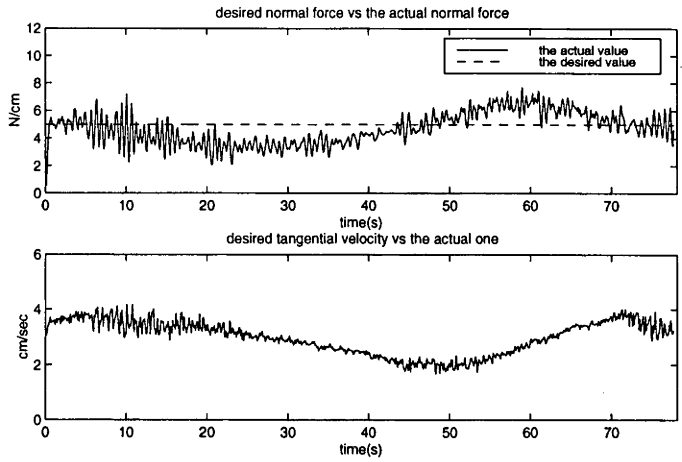


Figure 4-26: Experimental results: Normal force and tangential velocity with $\lambda = 0.97$

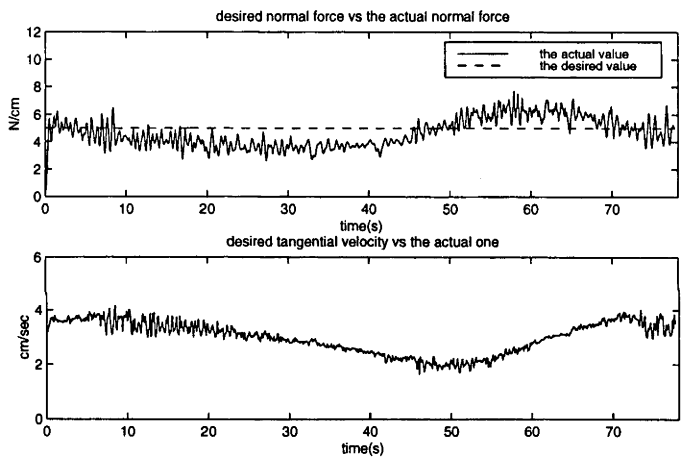


Figure 4-27: Experimental results: Normal force and tangential velocity with $\lambda = 0.99$

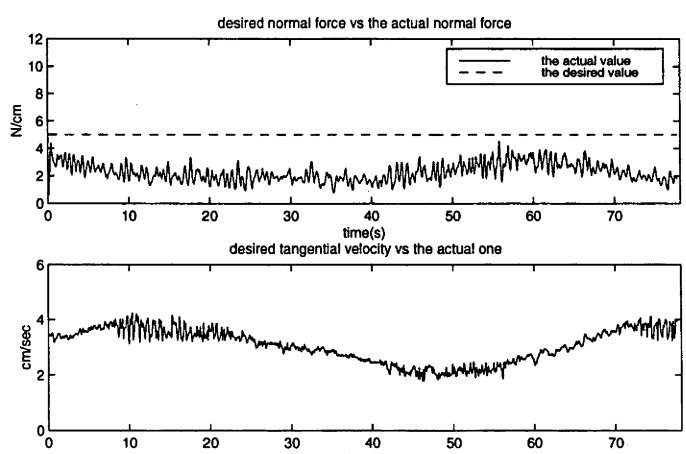


Figure 4-28: Experimental results: normal force and tangential velocity without adaptation

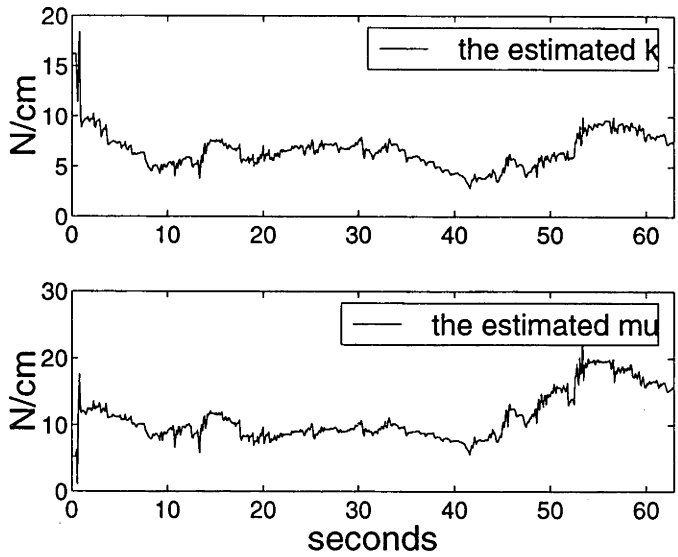


Figure 4-29: Experimental results: Parameter estimation with noise added

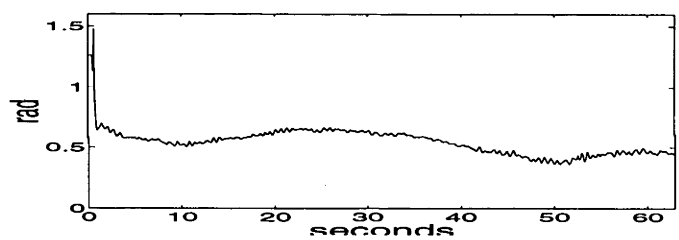


Figure 4-30: Experimental results: Angle estimation with noise added

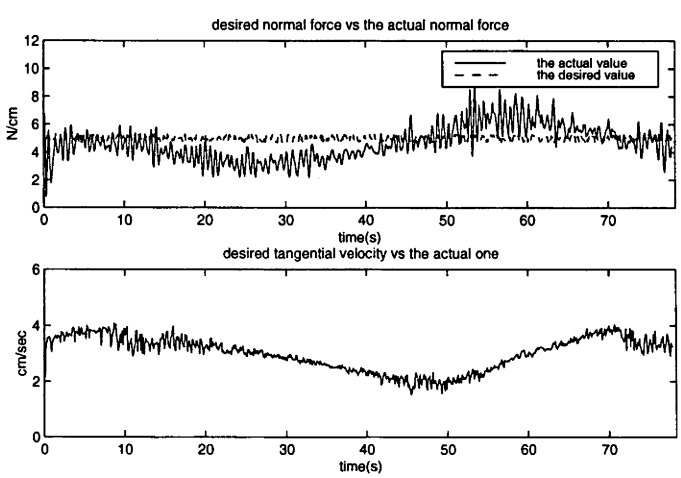


Figure 4-31: Experimental results: normal force and tangential velocity with noise added

illustrate that the proposed IAC approach is feasible and the performance is superior to that without using adaptation mechanism. Since adaptive control is more complex than normal conventional feedback control, the feedback control will be preferable provided that we can get a good approximation of the model parameters in advance. For example, if we know $\hat{\alpha}$ is around between 0.4 and 0.6 rad, it is not necessary to use adaptive control approach for our contour following control.

Chapter 5

Direct Velocity Control Experimental Results

5.1 Introduction

In this chapter, we present some experimental results related to the direct velocity control. The reason we treat this part as an independent chapter is that we did these experiments when my thesis was under review. We did some tuning work for the velocity control of the SCARA2 robot. This made the implementation of direct velocity control possible. Since direct velocity control rather than velocity-based position control is exploited, both force and velocity tracking are quite satisfactory.

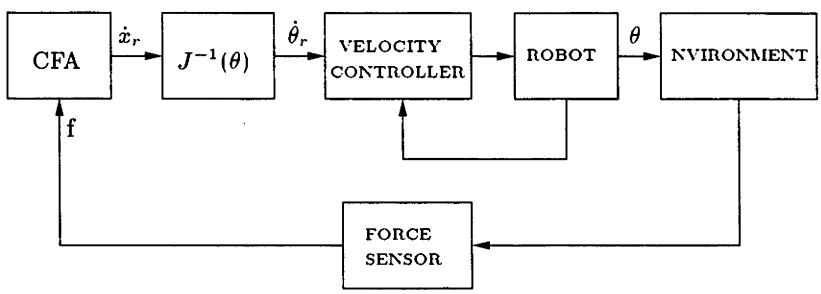


Figure 5-1: Block scheme of velocity control. CFA is contour following algorithm, $\dot{x}_r$ and $\dot{\theta}_r$ are the desired Cartesian and joint space velocities respectively, and f is the sensed force

5.2 The Implementation of Direct Velocity Control

The block scheme of the direct velocity control was shown in figure 3-4. For quick reference, we re-show this block scheme in figure 5-1.

5.3 Experimental Results

Firstly, we did experiments by use of the assumed known interaction model parameters. Figures 5-2,5-3 show the results when the gravel planar shape is an arc segment. When the peanut-shaped shape is used, the results are given in figures 5-4,5-5.

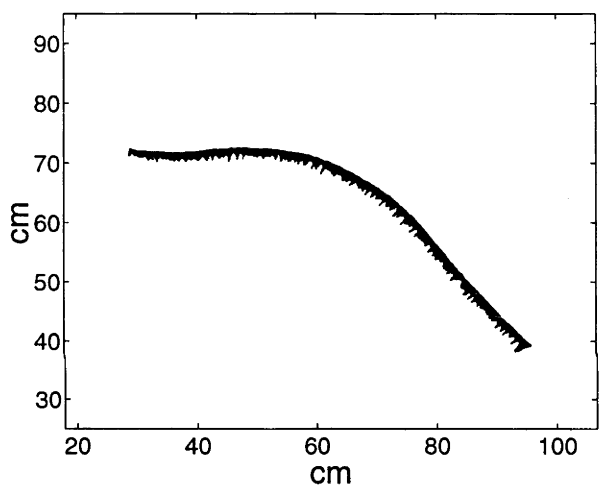


Figure 5-2: Actual arc contour and normal estimate

Next we give the experimental results as an indirect adaptive control approach is utilized. The results are shown in figures 5-6–5-8. For comparison, we also present the force and the velocity in figure 5-9 when adaptation is not used and the angle α is quite different from the actual value.

The experimental results demonstrate that when the interaction model parameters are known, the control algorithm can successfully realize the tracking of both the force and velocity. In addition, the algorithm keeps effective under different object shapes. If the model parameters are not known in advance, the proposed IAC approach can be used to reach the combination of the on-line parameter estimation and feedback control. Since the parameters can be quickly identified, the results of IAC method are similar to those when the almost true model parameters are used. The IAC method

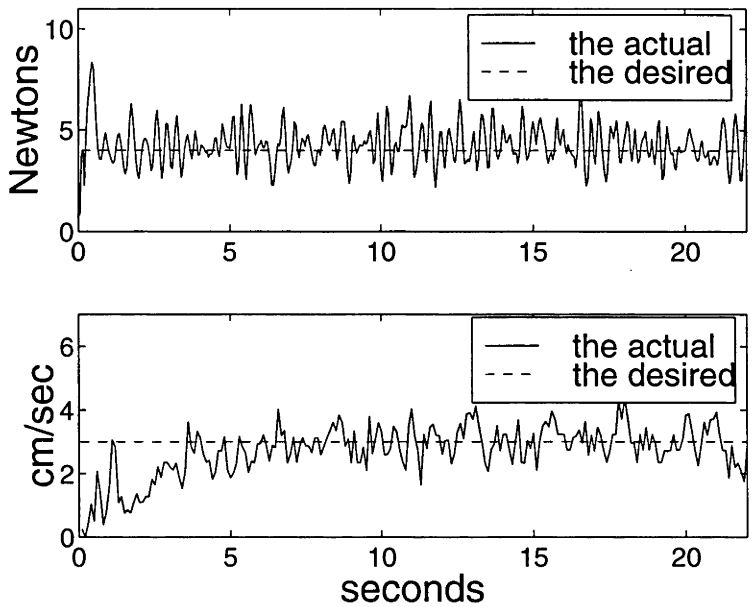


Figure 5-3: Normal forces and tangential velocities

is superior to the non-adaptation approach when the assumed model parameters are quite different from the true values as shown in figure 5-9.

5.4 Conclusion

By tuning the velocity controller parameters, we implemented the direct velocity control. Compared to the velocity-based position control, the direct velocity control can realize the tracking of both the force and velocity. Besides, the direct velocity control scheme is simpler.

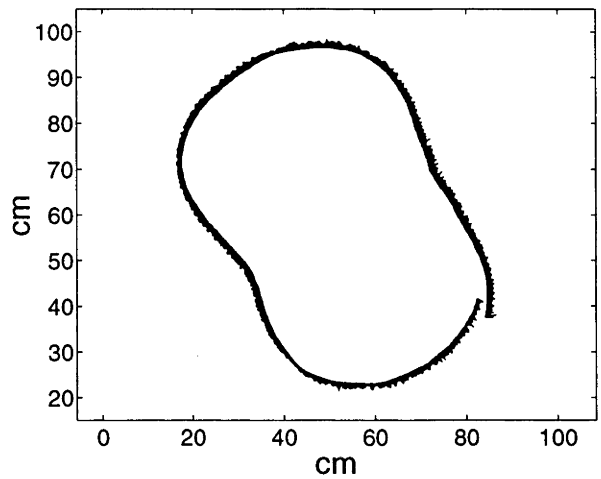


Figure 5-4: Actual contour shape and normal estimate

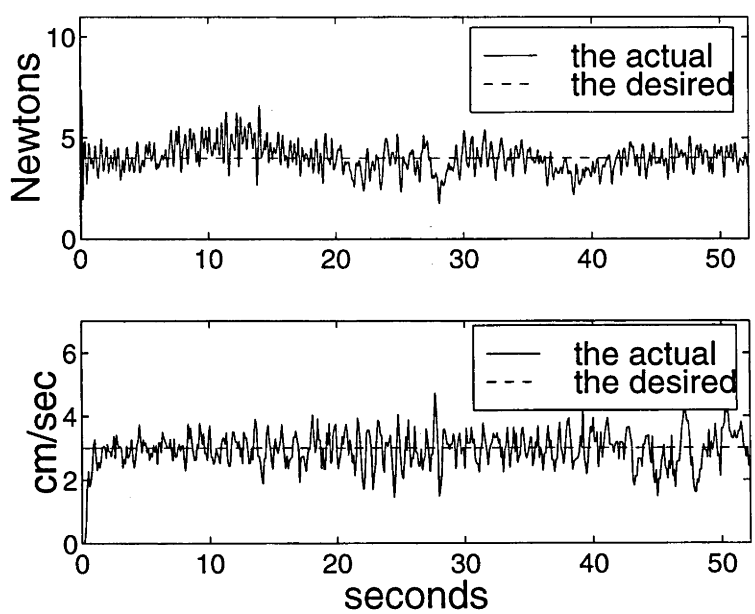


Figure 5-5: Normal forces and tangential velocities

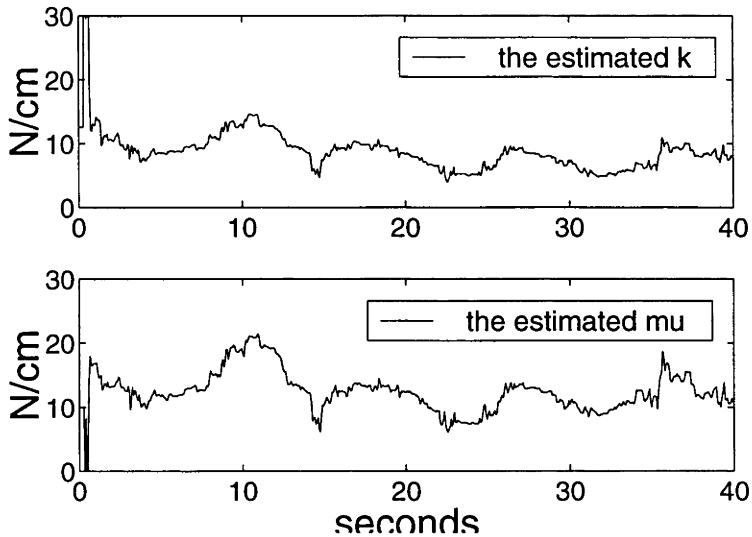


Figure 5-6: Parameter estimation with $\lambda = 0.97$

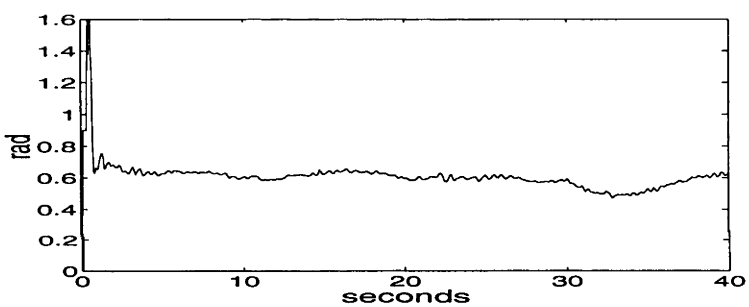


Figure 5-7: Estimate of α with $\lambda = 0.97$

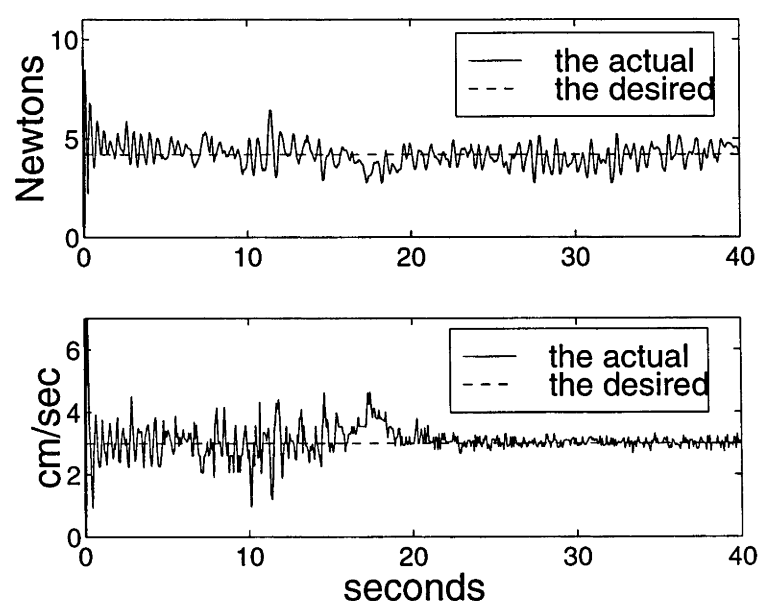


Figure 5-8: Normal forces and velocities with $\lambda = 0.97$

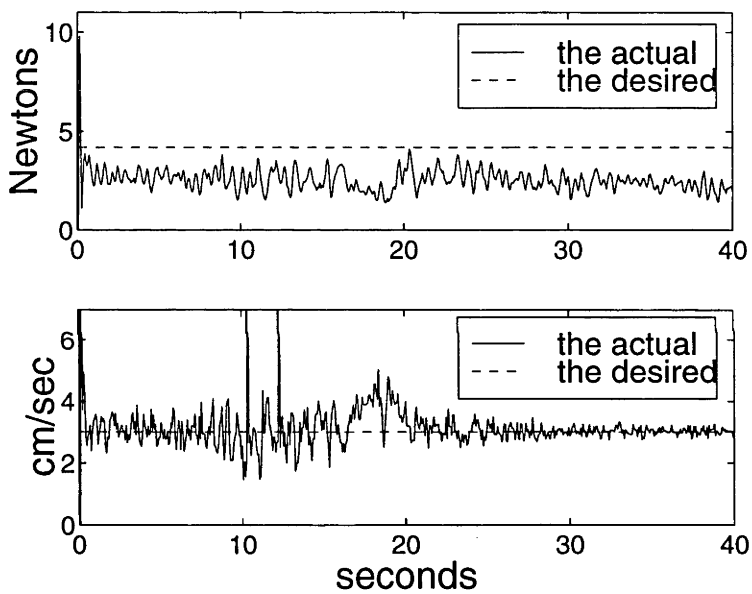


Figure 5-9: Forces and velocities without adaptation

Chapter 6

Conclusions

6.1 Conclusions

In this thesis, we have dealt with four problems related to manipulating a robot tool to move over the planar contour of the pile of gravel as followings:

1. Modelling the force-motion interactions between the robot tool and the gravel.
2. Velocity-based position control system to realize the tool contour following movement.
3. An indirect adaptive control algorithm to cope with the uncertainties in the object model parameters.
4. Direct velocity control approach.

There follows a summary of the results and conclusions for each of these problems.

Modelling the Force-Motion Interactions between the Robot Tool and the Gravel.

In this work, we concentrated on obtaining a model to characterise the force-motion interactions between the spheric tool and the gravel. It is novel since so far as we know, there is no similar research done before.

- Based on the initial experimental results, an empirical model is proposed to describe the interactions between the tool and the gravel. This model is quite

simple with only two parameters. The simple model structure is instrumental in the design of the controller.

- The model parameters are obtained off-line by use of least squares method. This model can be used to determine the contact depth and the surface normal or tangential when the tool touches the surface with a straightforward movement. By examining the estimates of the depth and the normal, the model has been verified to be suitable.

Velocity-Based Position Control System to Realize the Tool Contour Following Movement

In this work, we investigated how to let the spherical tool move over the planar contour of the pile of gravel. This is quite interesting and as far as we know, unique.

- A contour following algorithm(CFA) is developed. It combines two objectives: the constant tangential velocity and the desired normal contact force.
- With the consideration of the actual situation, we proposed a velocity-based position controller.
- A lot of simulations and experiments were conducted with different controller parameters, different model parameter estimates, and different contour shapes. The results demonstrate that the proposed controller is satisfactory.
- By slightly modifying the CFA, the tool can move along the contour in clockwise as well as counter clockwise(CCW) direction.

An Indirect Adaptive Control Algorithm to Cope with the Uncertainties in the Object Model Parameters

In this work, we focussed on deriving an indirect adaptive control algorithm to deal with the uncertainties in the interaction model parameters.

- A new description of the system input and output is deduced which is suitable for parameter estimation on line as well as off line.

- The combination of parameter estimation on-line and force feedback control brings about an IAC algorithm.
- Simulations and experiments were conducted with the IAC algorithm. The results show that the proposed IAC is feasible.

Direct Velocity Control Approach

After successfully tuning the velocity controller, we directly implemented the velocity control instead of velocity-based position control. Some experiments were conducted. The results demonstrate that it is superior to the velocity-based position control method.

6.2 Future Research

There are four main tasks to be done in the future. These tasks are:

- perceiving the granular materials, i.e. distinguishing granular materials from other objects by use of force sensing alone;
- extending the available control algorithm to deal with three-dimensional surfaces;
- exploiting manipulator dynamics to realize advanced robotic control, such as adaptive hybrid control and adaptive impedance control;
- finally realizing the contour following movement and vacuum cleaning the gravel simultaneously.

We can imagine that the granular materials in the real world can be heaped in various shapes. Many of them actually do not have any regular shape. Maybe it is just a thick layer of material. So the control system should be designed to deal with all possible situations met in practical applications.

Bibliography

[1] S. Arimoto, Y. Liu, and T. Naniwa. Model-based adaptive hybrid control for geometrically constrained robots. In *Proceedings of IEEE International Conference on Robotics and Automation*, Atlanta, 1993.

[2] K.J. Astrom and B. Wittenmark. *Adaptive Control.* Addison Wesley, Reading, Mass., 1989.

[3] M. Blauer and P.R Belanger. State and parameter estimation for robotic manipulators using force measurements. In *IEEE Transactions on Automatic Control,* volume AC-32, December 1987.

[4] N.K. Bose. *Digital Filters : Theory and Applications.* Elsevier Science Pub. Co. 1985

[5] D. Bossert, Uy-Ly and J. Vagners. Experimental evaluation of a hybrid position and surface following algorithm for unknown surfaces. In *Processings of the 1996 IEEE International Conference on Robotics and Automation*, pages 2252-2257, April 1996.

[6] S. Chiaverini, B. Siciliano and L. Villani. Force/position regulation of compliant robot manipulators. In *IEEE Transactions on Robotics and Automation*, volume 39, pages 647-652, 1994.

[7] S. Chiaverini, B. Siciliano and L. Villani. Force and position tracking: parallel control with stiffness adaptation. In *IEEE Control Systems*, volume 18(1), pages 27-33, February 1998.

[8] R. Colbaugh, H. Seraji and K. Glass. Direct adaptive impedance control of robot manipulators. In *Journal of Robotic Systems*, volume 10(2), pages 217-248, 1993.

[9] R. Colaugh, K. Glass and H. Seraji. Direct adaptive control of robotic systems. In *Proceedings of American Control Conference*, pages 1138-1144, San Francisco, 1993.

[10] R. Colbaugh, H. Seraji, and K. Glass. Adaptive compliant motion control for dexterous manipulators. In *International Journal of Robotics Research*, 14(3), pages 270-280, 1995.

[11] J. Craig, P. Hsu and S. Sastry. Adaptive control of mechanical manipulator. In *International Journal of Robotics Research*, volume 6(2), pages 16-28, 1987.

[12] Y. Ekalo and M. Vukobratovic. Robust and adaptive position/force stabilisation of robotic manipulators in contact task. In *Robotica*, volume 11(4), pages 373-386, 1993.

[13] Y. Ekalo and M. Vukobratovic. Adaptive stabilisation of motion and forces in contact tasks for robotic manipulators with non-stationary dynamics. In *International Journal of Robotics and Automation*, volume 9(3), pages 91-98, 1994.

[14] R. Goddard, Y-F. Zheng and H. Hemami. Dynamic Hybrid velocity/force control of robot compliant motion over globally unknown objects. In *IEEE Transactions on on Robotics and Automation*, Volume 8(1), pages 132-138, February 1992.

[15] G.C. Goodwin and K.S. Sin. *Adaptive Filtering Prediction and Control.* Information and System Science Series. Englewood Cliffs. N. J.: Prentice-Hall. 1984.

[16] H. Hemami and R.E. Godard. Recognition of geometric shape by a robotic system. In *Journal of Robotic Systems*, volume 4(2), pages 237-257, April 1987.

[17] H. Hemami, J. Bay and R.E. Godard. A conceptual framework for tactually guided exploration and shape perception. In *IEEE Transactions on Biomedical Engineering*, volume 35(2), pages 99-109, February, 1988.

[18] N. Hogan. Impedance control: an approach to manipulation. In *ASME Journal Dynamic System, Measurement and Control,* volume 107, pages 1-24, 1985.

[19] N. Hogan. On the stability of manipulators performing contact tasks. In *IEEE Journal of Robotics and Automation*, volume 4, pages 677-686, 1988.

[20] H.M. Jaeger, S.R. Nagel and R.P. Behringer. The Physics of Granular Materials. In *Physics Today*, pages 32-39, April 1996.

[21] J. Jean and L. Fu. Efficient adaptive hybrid control strategies for robot in constrained manipulation. In *IEEE International Conference on Robotics and Automation*, Sacramento, pages 1682-1686, 1991.

[22] P. Kazanides, N.S. Bradley and W.A. Wolovich. Dual-drive force/velocity control: Implementation and experimental results. In *IEEE International Conference on on Robotics and Automation,* pages92-97, May 1989.

[23] D. Kestenbaum. Sand castles and cocktail nuts. In *New Scientist*, pages 25-28, May 1997.

[24] W. Li and J. Slotine. Indirect adaptive robot control. In *5th IEEE International Conference on Robotics and Automation*, Philadelphia, 1988.

[25] R. Lozano and B. Brogliato. Adaptive hybrid force/position control for redundant manipulators. In *IEEE Transactions on Automatic Control*, volume 37(10), pages 1501-1505, 1992.

[26] W. Lu and Q. Meng. Impedance control with adaptation for robot manipulators. In *IEEE Transactions on Robotics and Automation*, volume 7, pages 408-415, 1991.

[27] J.P. Merlet C-surface applied to the design of a hybrid force-position robot controller. In *Proceedings of the 1987 IEEE International Conference on on Robotics and Automation,* Raleigh, North Carolina, volume 2, pages 1055-1059, April 1987.

[28] R.H. Middleton and G.C. Goodwin. Adaptive computed-torque control for rigid-link manipulators. In *System Control Letters*, volume 10, pages 9-16, 1988.

[29] J.K. Mills and D.M. Lokhorst. Control of robotic manipulators during general task execution: a discontinuous control approach. In *International Journal of Robotics Research*, volume 12, pages 146-163, 1993.

[30] K. Narendra and A. Annaswamy. In *Stable Adaptive System*, Prentice Hall, Englewood Cliffs, NJ, 1989.

[31] U. Nunes, P. Faia and A.T. Dealmeida. Sensor-based 3-D autonomous surface-following control. In *Mathematics and computers in simulation,* volume 41(5)-(6), pages 429-444, August 1996.

[32] S.R. Pandian and S. Kawamura. Hybrid force position control for robot manipulators based on a D-type learning law. In *Robotica,* volume 14(1), pages 51-59, January-February 1996.

[33] Y.V. Podurayea. Contour control and motion modelling for robotic manipulators based on dynamic models. In *Journal of computer and systems sciences international,* volume 33(1), pages 150-158, January-February 1995.

[34] J. Reed and P. Ioannou. Instability analysis and robust adaptive control of robotic manipulators. In *IEEE Transactions on Robotics and Automation*, volume 5(3), pages 383-386, 1989.

[35] M. Raibert and J. Craig. Hybrid position/force control for manipulators. In *ASME Journal of Dynamic System, Measurement and Control*, volume 103, pages 126-133, 1981.

[36] S.K. Singh and D.O. Popa. An analysis of some fundamental problems in adaptive control of force and impedance behaviour: Theory and experiments. In *IEEE Transactions on Robotics and Automation*, volume 11(6), . pages 912-921, 1995.

[37] B. Siciliano and L. Villani. Adaptive compliant control of robot manipulators. In *Control Engineering Practice,* 4(5), pages 705-712, 1996.

[38] J.J. Slotine and W. Li. On the adaptive control of robot manipulators. In *International Journal of Robotics Research*, volume 6(3), pages 49-59, 1987.

[39] J.J. Slotine and W. Li. Composite adaptive control of robot manipulators. In *Automatica*, volume 25(4), pages 509-519, 1989.

[40] J.J. Slotine and W. Li. Adaptive Strategies in constrained manipulators. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 595-601, Raleigh, 1987.

[41] T. Soderstrom and P. Stoica. *System Identification.* Systems and Control Engineering Series. Englewood Cliffs NJ Prentice-Hall. 1989.

[42] C. Stefano and S. Lorenzo. The parallel approach to force/position control of robotic manipulators. In *IEEE Transactions on Robotics and Automation*, volume 9(4), pages 361-373, 1993.

[43] C. Su and Y. Steponenko. Hybrid adaptive/robust motion control of rigid-link electrically-driven robot manipulators. In *IEEE Transactions on Robotics and Automation*, volume 11(3), pages 426-432,1995.

[44] C.S. Tzafestas, N.K. M'sirdi and N. Manamani. Adaptive impedance control applied to a pneumatic legged robot. In *Journal of Intelligent and robotic systems*, volume 20, pages 105-129, October-December 1997.

[45] M.K. Vukobratovic and Y. Ekalo. New approach to control of robotic manipulators interacting with dynamic environment. In *Robotica*, volume 14, part I, pages 31-39, 1996.

[46] M. Vukobratovic and A. Tuneski. Adaptive control of single rigid robotic manipulators interacting with dynamic environment - an overview. In *Journal of Intelligent and Robotic Systems,* volume 17, pages 1-30, 1996.

[47] K. Wedeward, R. Colbaugh, and A. Engelmann. Adaptive explicit force control of position-controlled manipulators. In *Journal of Robotic System*, volume 13(9), pages 603-618, 1996.

[48] L.L. Whitcomb, S. Arimoto, T. Naniwa, and F. Ozaki. Adaptive model-based hybrid control of geometrically constrained robot arms. In *IEEE Transactions on Robotics and Automation*, volume 13(1), pages 105-116, 1997.

[49] C.C. Wit and B. Brogliato. Direct adaptive impedance control including transition phases. In *Automatica*, volume 33(4), pages 643-649, 1997.

[50] J. Wu, Z. Luo, M. Yamakita and K. Ito. Adaptive hybrid control of manipulators on uncertain flexible objects. In *Advanced Robotics*, volume 10(5), pages 469-485, 1996.

[51] B. Yao and M. Tomizuka. Adaptive control of robot manipulators in constrained motion. In *Proceedings of American Control Conference*, pages 1128-1132, San Francisco, 1993.

[52] B. Yao and M. Tomizuka. Robust adaptive constrained motion and force control of manipulators with guaranteed transient performance. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 893-898, Nagoya, 1995.

[53] B. Yong. Contour following of a force-controlled industrial robot using preview control. In *Ksme Journal,* volume 11(2), pages 115-124, April 1997.

[54] T. Yoshikawa and A. Sudou. Dynamic hybrid position/force control of robot mianipulators-on line estimation. In *IEEE Transactions on Robotics and Automation*, volume 9(2), pages 220-226, April 1993.

[55] J. Yuan. Composite adaptive control of constrained robots. In *IEEE Transactions on Robotics and Automation*, volume 12(4), pages 640-645, 1996.

[56] G. Zeng and A. Hemami. An overview of robot force control. In *Robotica,* volume 15, pages 473-482, 1997.

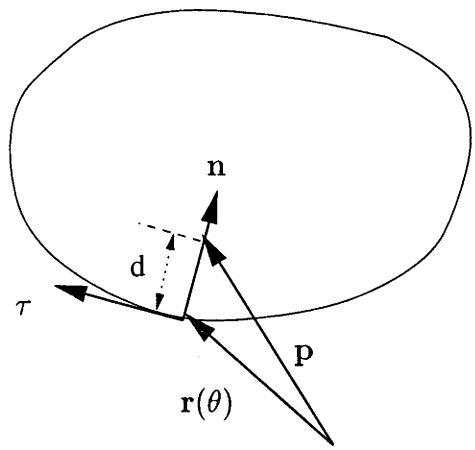
Appendix A

Computation of Next Tool Position

This appendix describes the control modifications needed if the tool is required to move along the pile contour in clockwise rather than CCW direction as shown in Figure A-1. When the tangential direction is chosen to be opposite to that as shown in figure 3-2, we set $\mathbf{e_f}$ and $\mathbf{e_g}$ as shown in figure A-2. So if the contour is closed loop, the tool will move clockwise. The two velocity vectors, $\mathbf{v_f}$ and $\mathbf{v_g}$, are still along $\mathbf{e_f}$ and $\mathbf{e_g}$, respectively. Equation (3.9) still can be used to calculate v_f and v_g. Using figure A-3, we have

$$\begin{cases} v_x = v_f \cos\eta - v_g \sin\eta \\ v_y = v_f \sin\eta + v_g \cos\eta \end{cases} \tag{A.1}$$

where $\eta = \theta_1 + \theta_2 + \beta$, $\beta = atan2(f_y, f_x)$. Then we get the new tool position by solving equations (3.16) and (3.17).



θ is a free parameter

Figure A-1: Tool movement direction

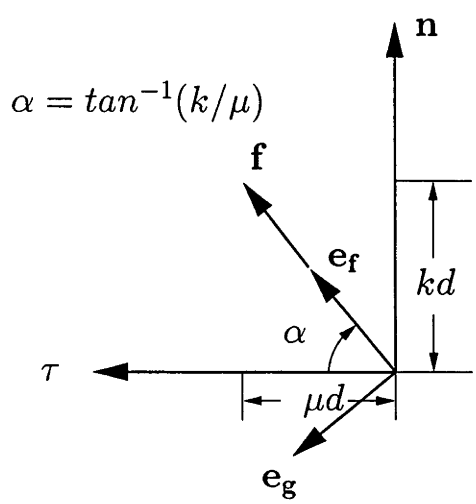


Figure A-2: Determination of the control velocities

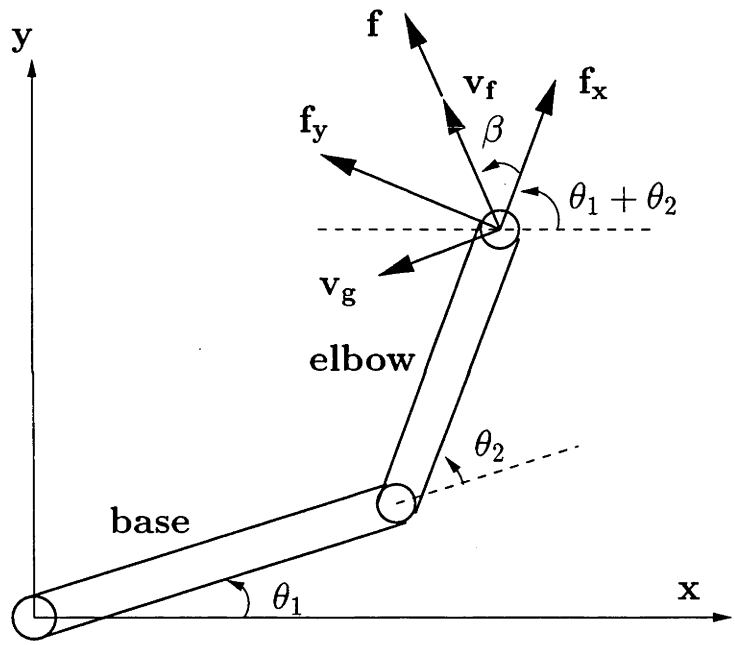


Figure A-3: Computation of new tool position using control velocities